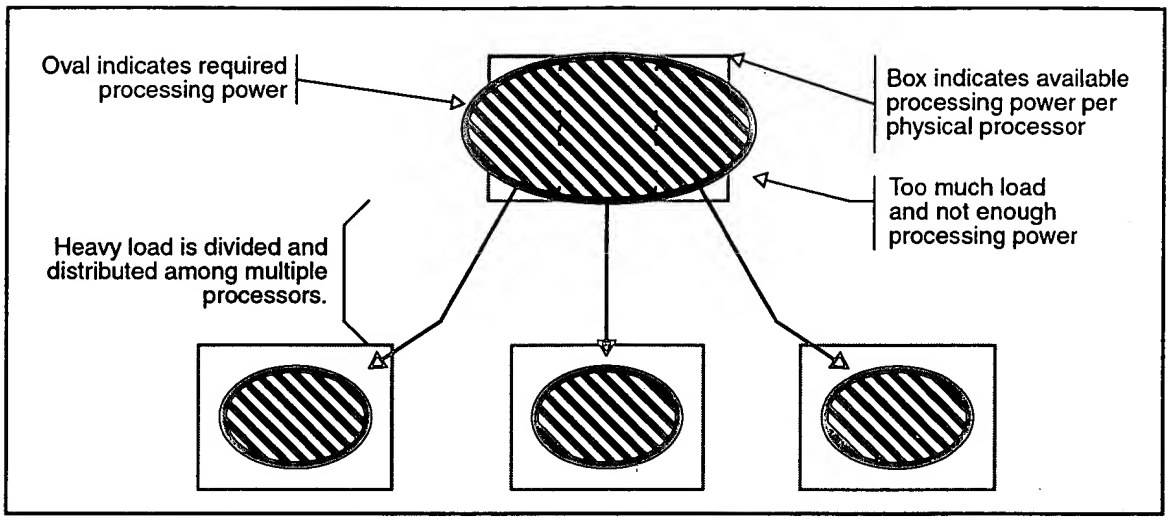

Figure 1

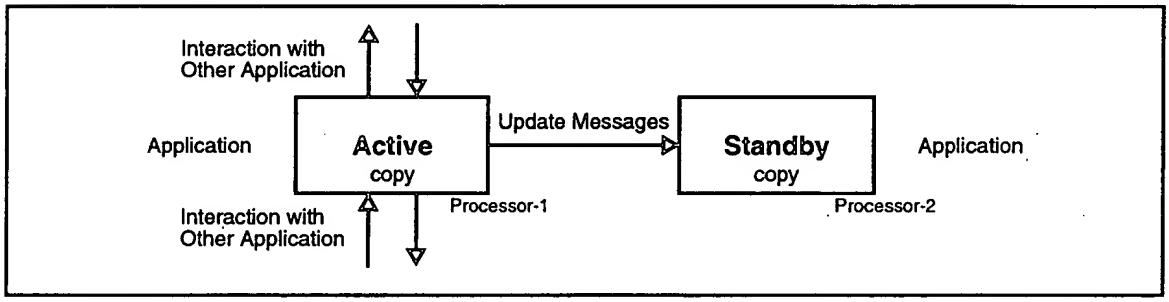

Figure 2

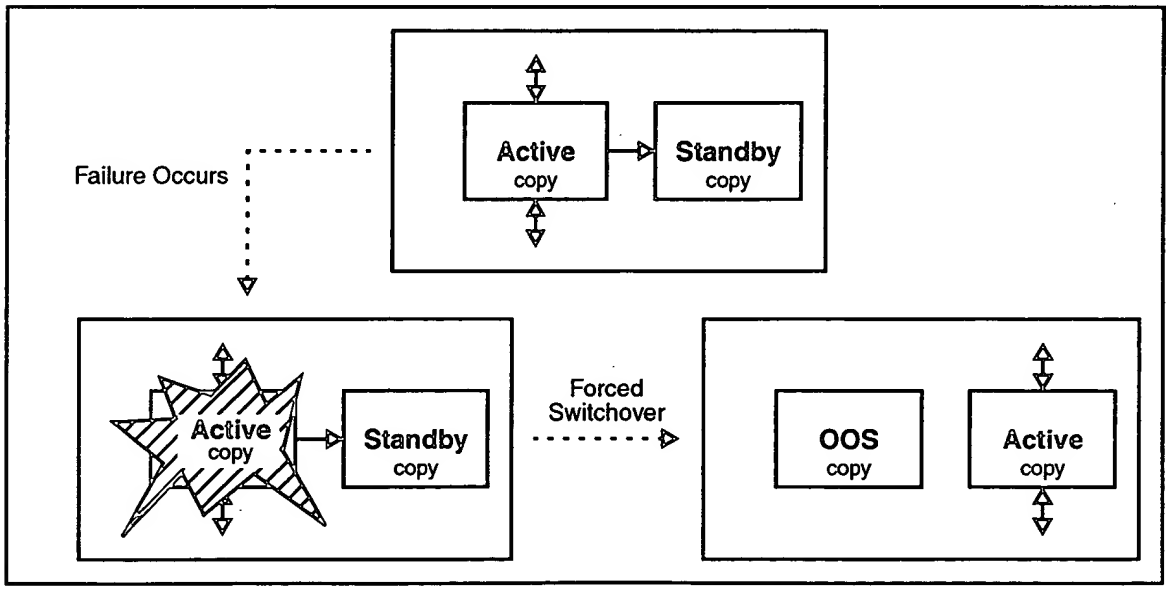

Figure 3

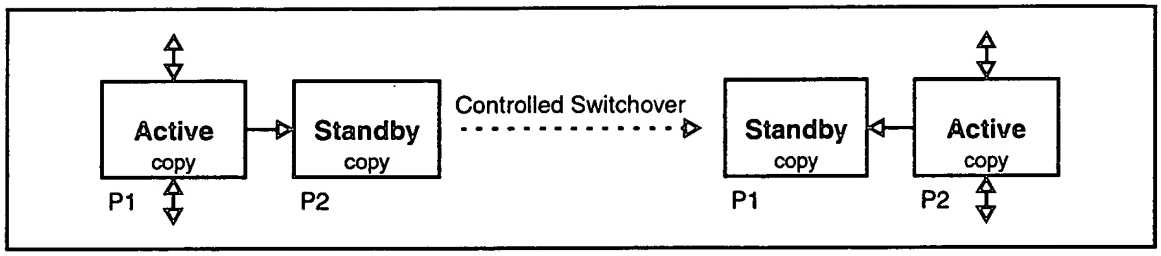

Figure 4

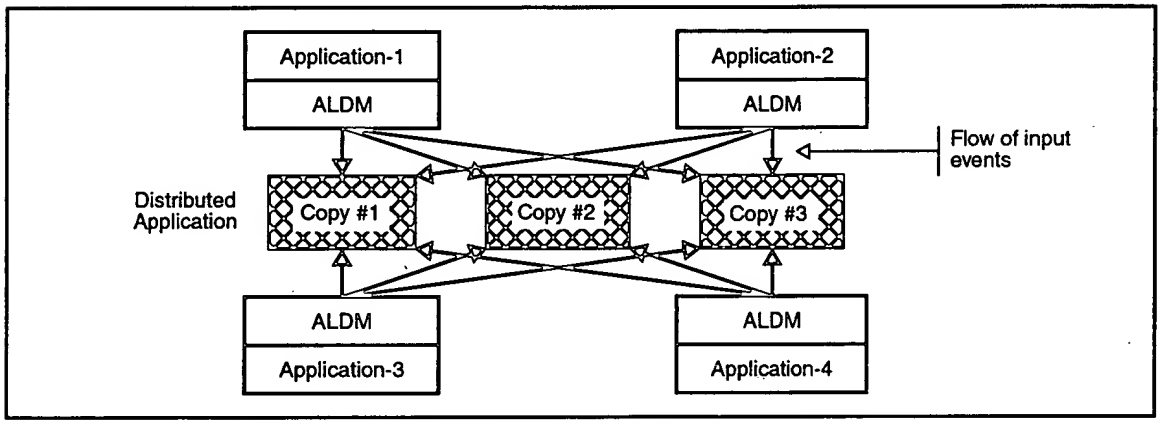

Figure 5

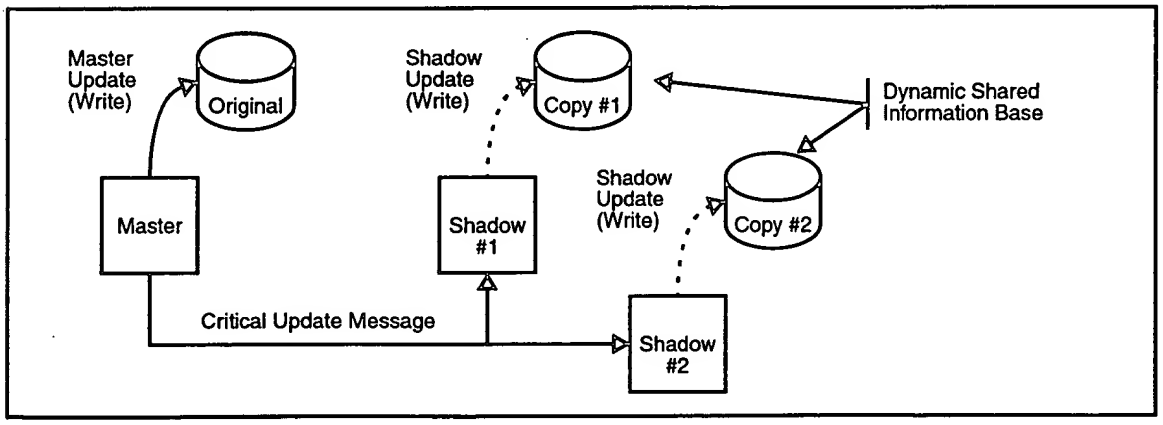

Figure 6

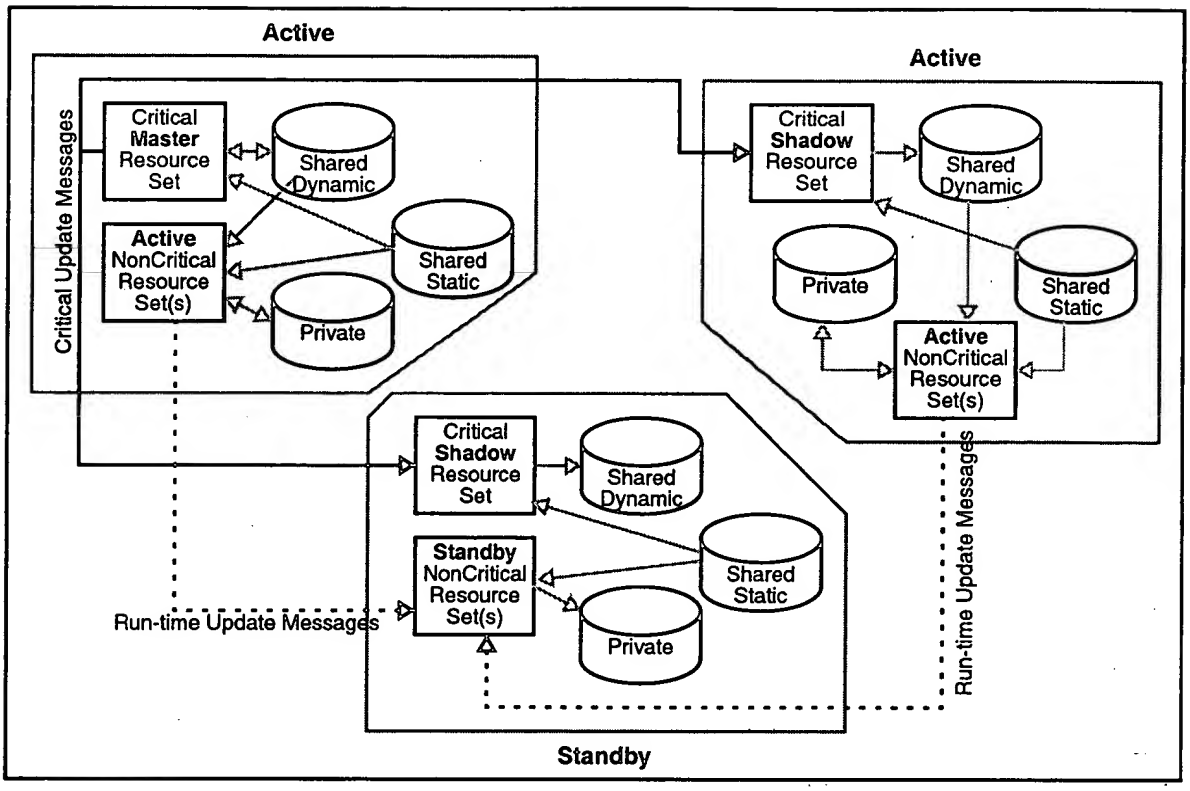

Figure 7

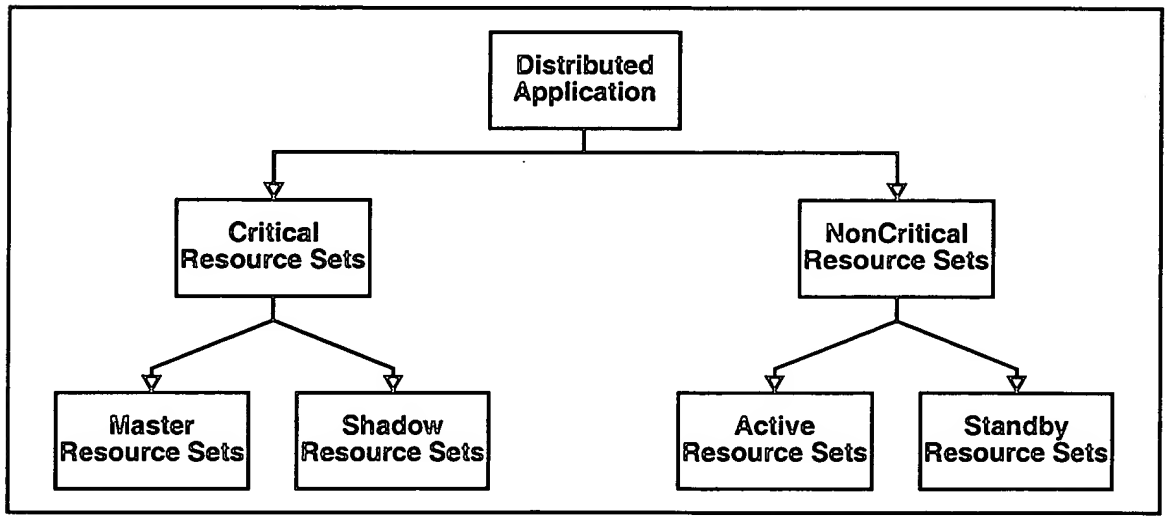

Figure 8

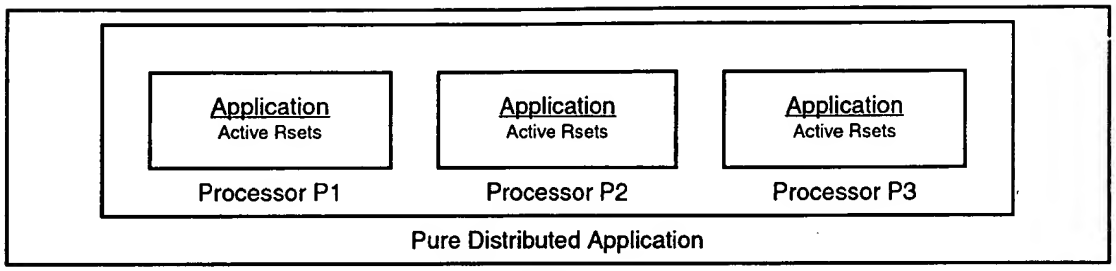

Figure 9

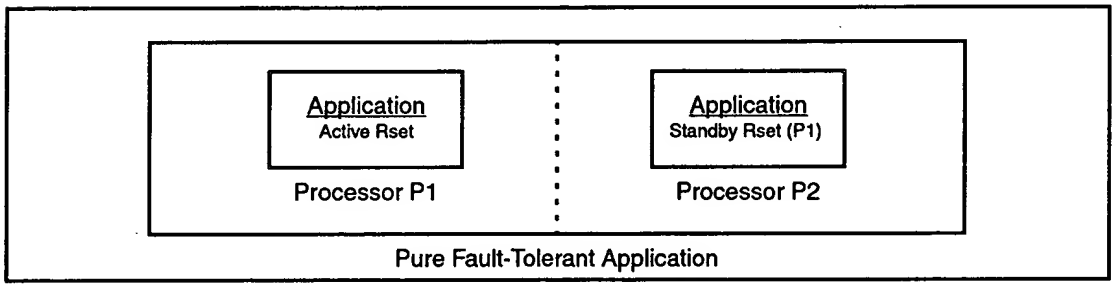

Figure 10

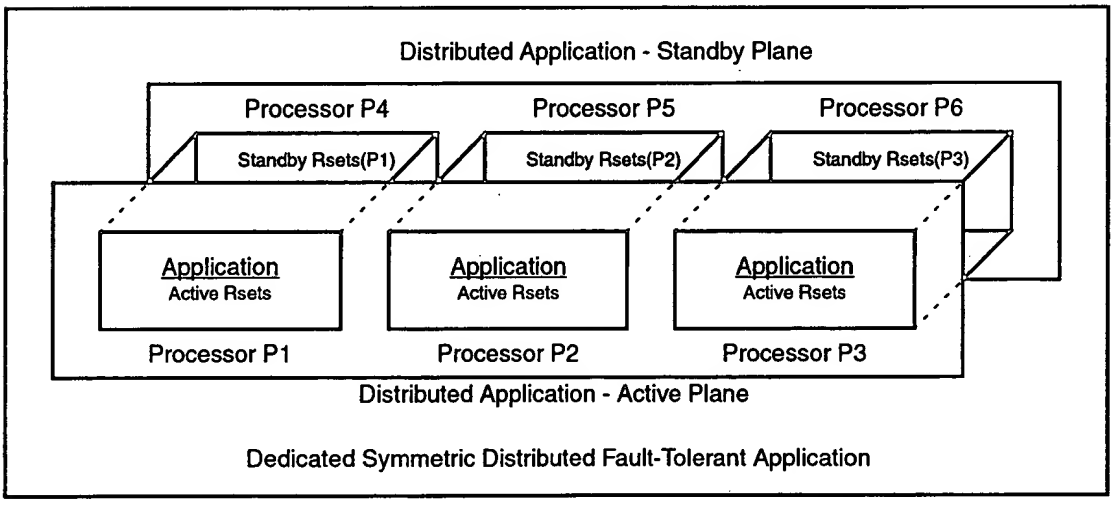

Figure 11

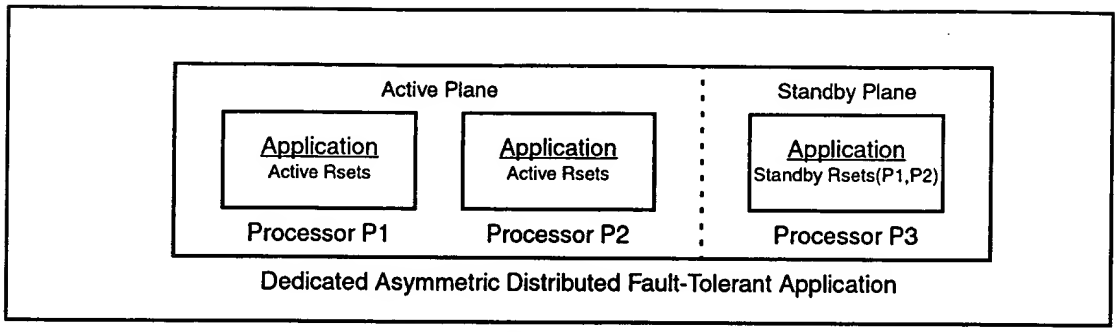

Figure 12

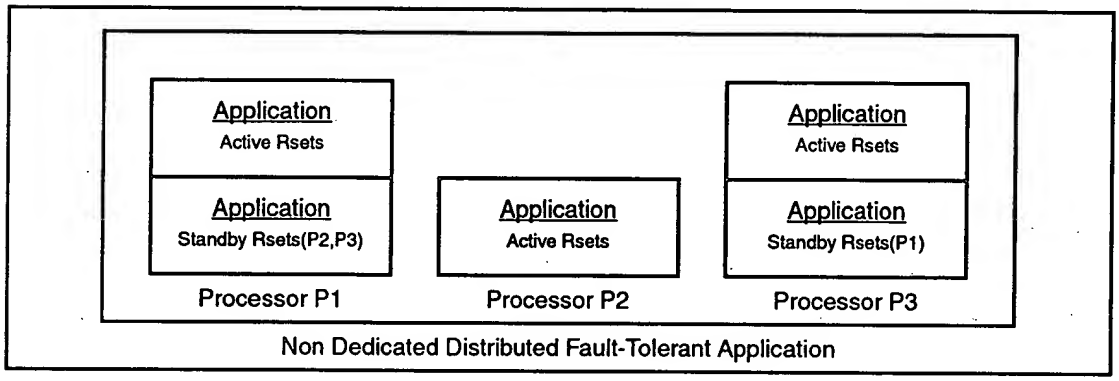

Figure 13

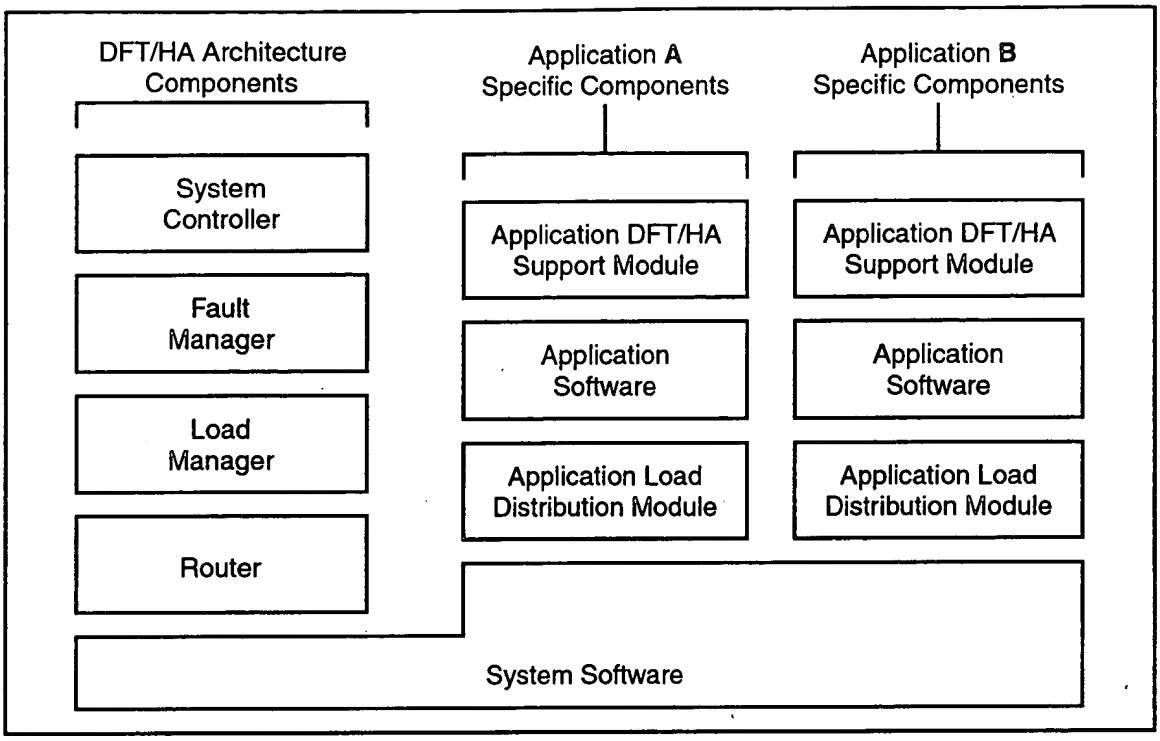
DFT/HA Architecture Components
Application A Specific Components
Application B Specific Components
System Controller
Fault Manager
Load Manager
Router
Application DFT/HA Support Module
Application Software
Application Load Distribution Module
Application DFT/HA Support Module
Application Software
Application Load Distribution Module
System Software

Figure 14

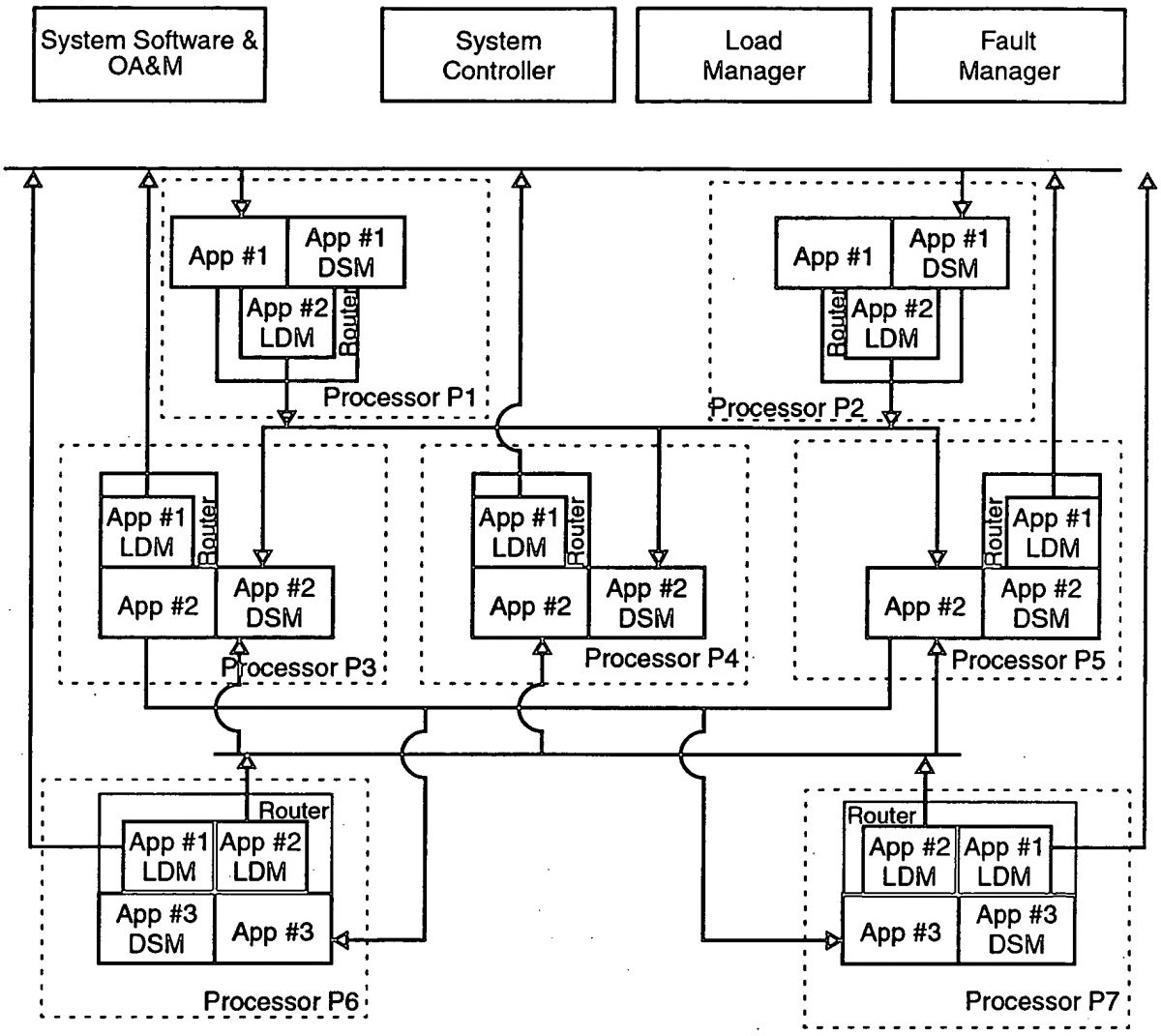
System Software & OA&M
System Controller
Load Manager
Fault Manager
App #1
App #1 DSM
App #2 LDM
Router
Processor P1
Processor P2
App #1 LDM
App #2
App #2 DSM
Processor P3
Processor P4
Processor P5
App #3 DSM
App #3
Processor P6
Processor P7

Figure 15

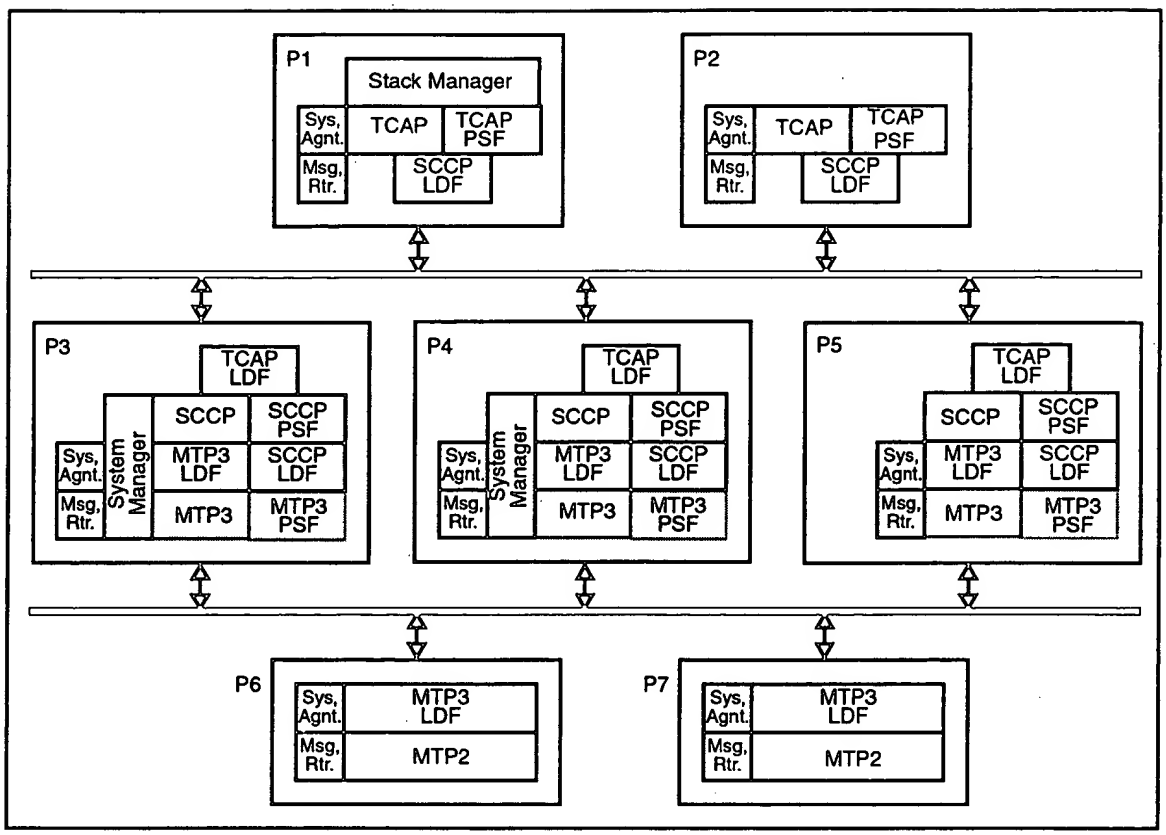
P1
Stack Manager
Sys, Agnt.
TCAP
TCAP PSF
Msg, Rtr.
SCCP LDF
P2
Sys, Agnt.
TCAP
TCAP PSF
Msg, Rtr.
SCCP LDF
P3
TCAP LDF
Sys, Agnt.
System Manager
SCCP
SCCP PSF
MTP3 LDF
SCCP LDF
Msg, Rtr.
MTP3
MTP3 PSF
P4
TCAP LDF
Sys, Agnt.
System Manager
SCCP
SCCP PSF
MTP3 LDF
SCCP LDF
Msg, Rtr.
MTP3
MTP3 PSF
P5
TCAP LDF
SCCP
SCCP PSF
Sys, Agnt.
MTP3 LDF
SCCP LDF
Msg, Rtr.
MTP3
MTP3 PSF
P6
Sys, Agnt.
MTP3 LDF
Msg, Rtr.
MTP2
P7
Sys, Agnt.
MTP3 LDF
Msg, Rtr.
MTP2

Figure 16

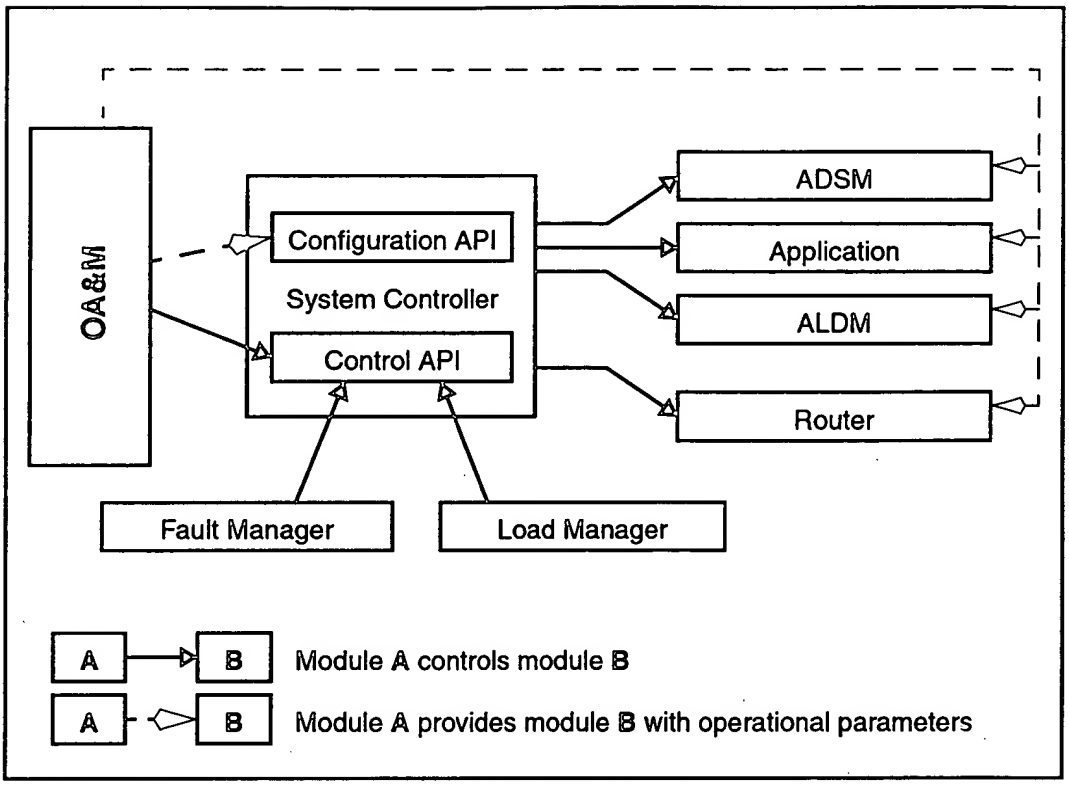

Figure 17

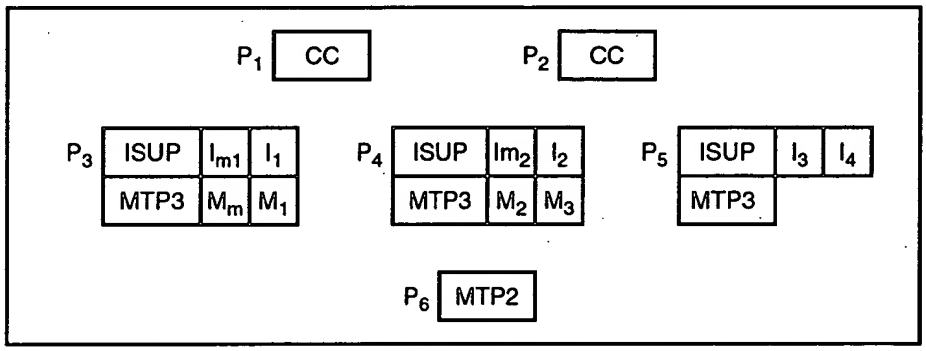

Figure 18

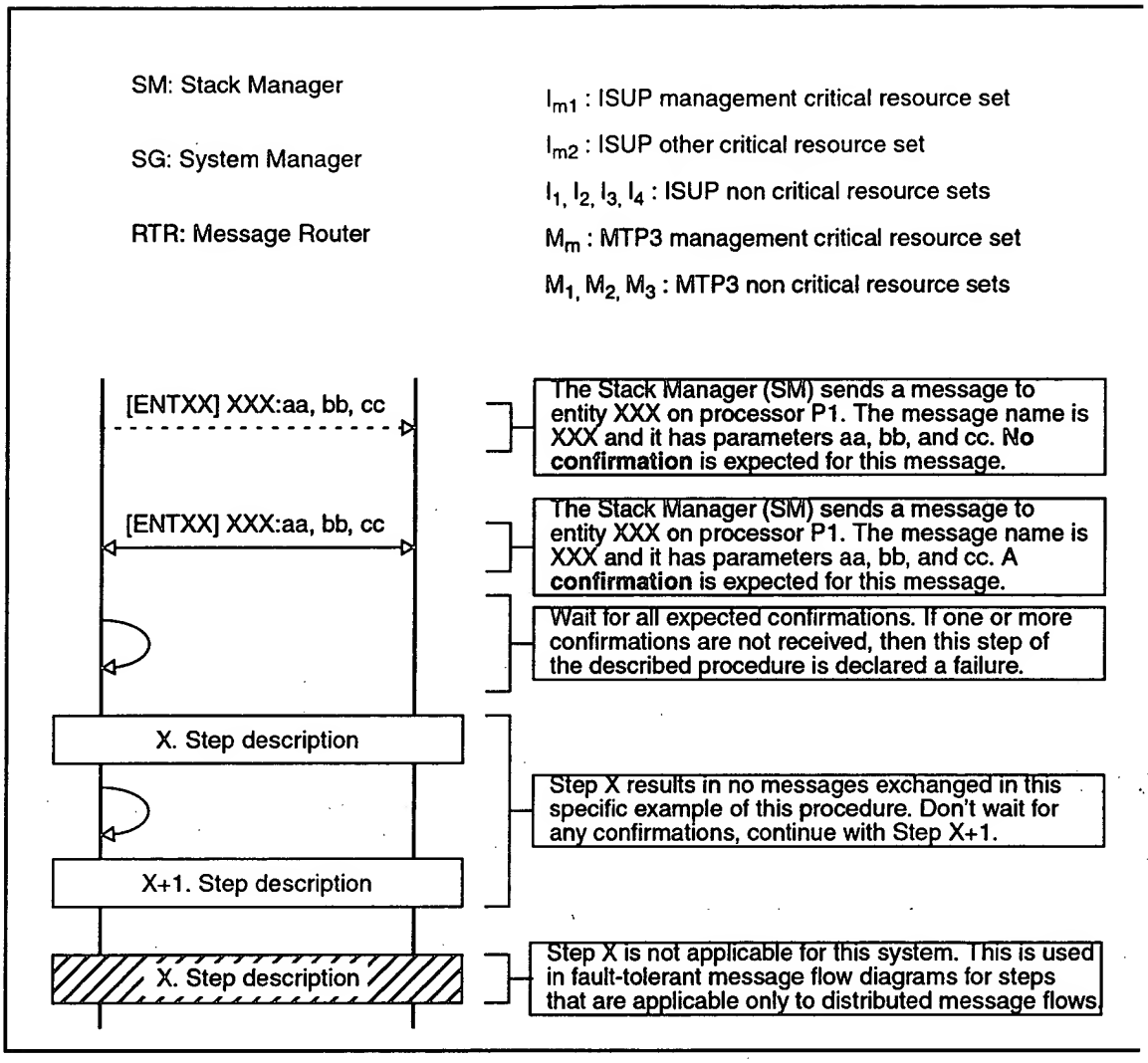
SM: Stack Manager
SG: System Manager
RTR: Message Router
I_{m1} : ISUP management critical resource set
I_{m2} : ISUP other critical resource set
I_1, I_2, I_3, I_4 : ISUP non critical resource sets
M_m : MTP3 management critical resource set
M_1, M_2, M_3 : MTP3 non critical resource sets
[ENTXX] XXX:aa, bb, cc
The Stack Manager (SM) sends a message to entity XXX on processor P1. The message name is XXX and it has parameters aa, bb, and cc. No confirmation is expected for this message.
[ENTXX] XXX:aa, bb, cc
The Stack Manager (SM) sends a message to entity XXX on processor P1. The message name is XXX and it has parameters aa, bb, and cc. A confirmation is expected for this message.
Wait for all expected confirmations. If one or more confirmations are not received, then this step of the described procedure is declared a failure.
X. Step description
Step X results in no messages exchanged in this specific example of this procedure. Don't wait for any confirmations, continue with Step X+1.
X+1. Step description
X. Step description
Step X is not applicable for this system. This is used in fault-tolerant message flow diagrams for steps that are applicable only to distributed message flows.

Figure 19

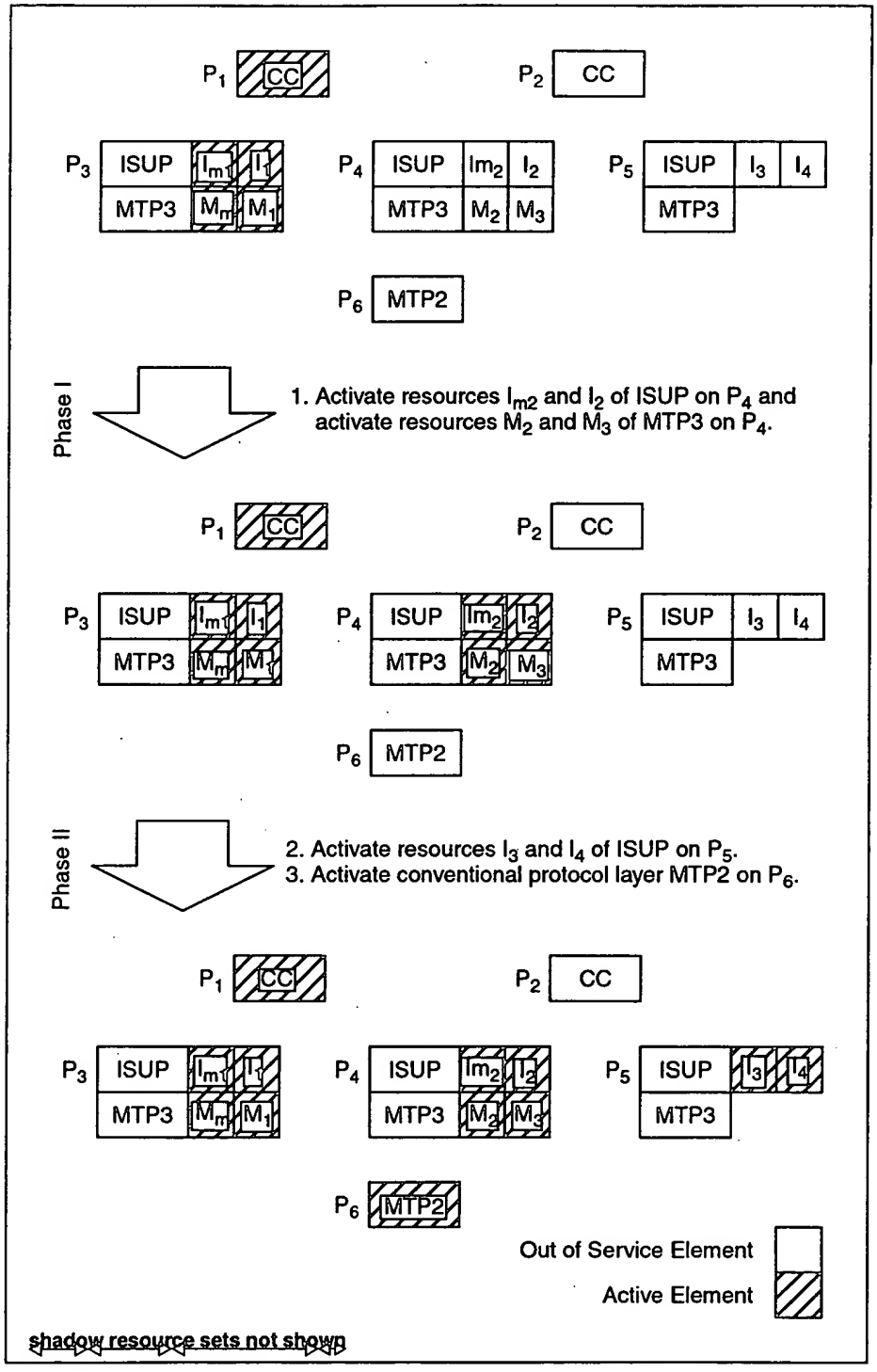
P1 CC
P2 CC
P3 ISUP Im1 I1
MTP3 Mm M1
P4 ISUP Im2 I2
MTP3 M2 M3
P5 ISUP I3 I4
MTP3
P6 MTP2
Phase I
1. Activate resources Im2 and I2 of ISUP on P4 and activate resources M2 and M3 of MTP3 on P4.
P1 CC
P2 CC
P3 ISUP Im1 I1
MTP3 Mm M1
P4 ISUP Im2 I2
MTP3 M2 M3
P5 ISUP I3 I4
MTP3
P6 MTP2
Phase II
2. Activate resources I3 and I4 of ISUP on P5.
3. Activate conventional protocol layer MTP2 on P6.
P1 CC
P2 CC
P3 ISUP Im1 I1
MTP3 Mm M1
P4 ISUP Im2 I2
MTP3 M2 M3
P5 ISUP I3 I4
MTP3
P6 MTP2
Out of Service Element
Active Element
shadow resource sets not shown

Figure 20

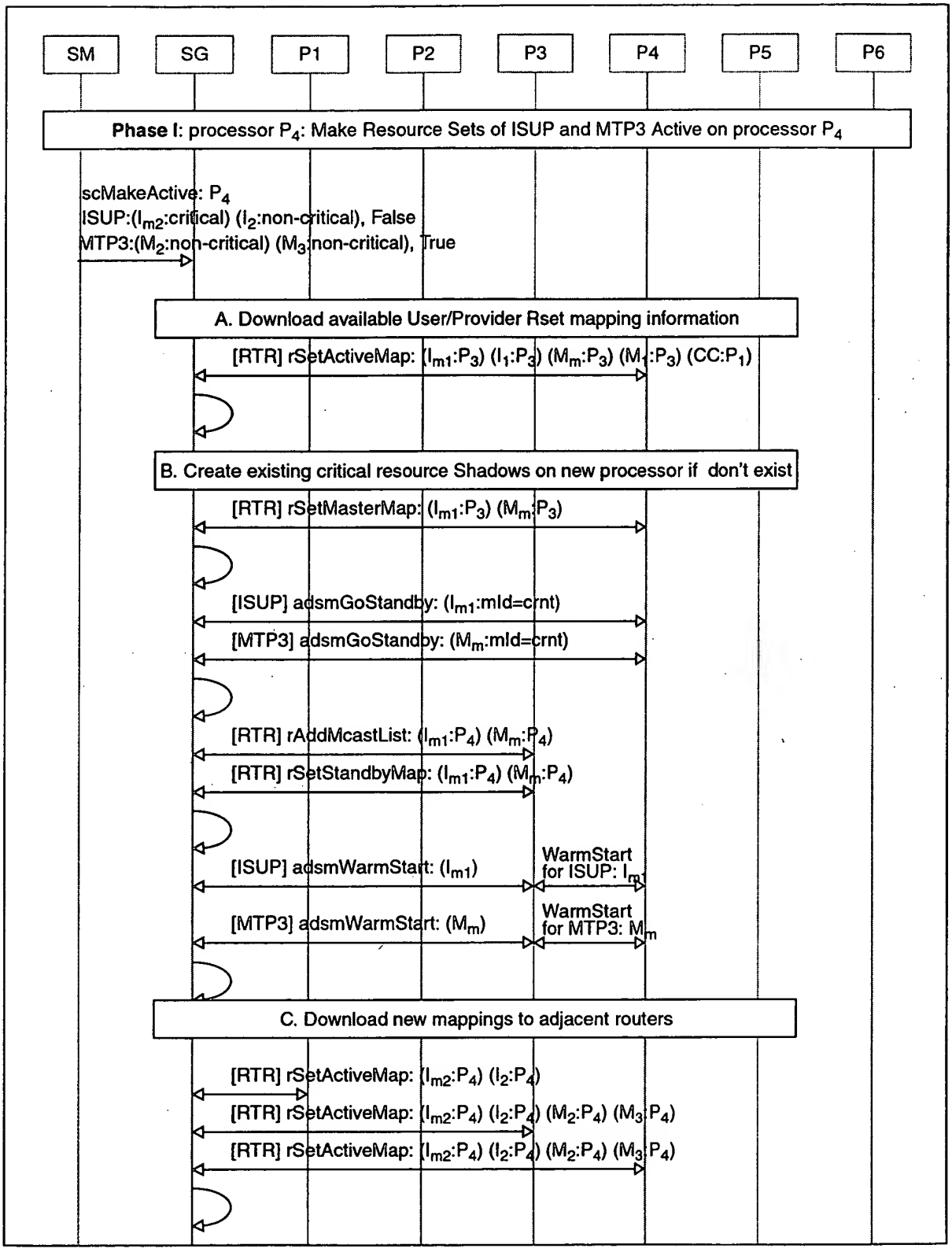
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: processor P4: Make Resource Sets of ISUP and MTP3 Active on processor P4
scMakeActive: P4
ISUP:(Im2:critical) (I2:non-critical), False
MTP3:(M2:non-critical) (M3:non-critical), True
A. Download available User/Provider Rset mapping information
[RTR] rSetActiveMap: (Im1:P3) (I1:P3) (Mm:P3) (M1:P3) (CC:P1)
B. Create existing critical resource Shadows on new processor if don't exist
[RTR] rSetMasterMap: (Im1:P3) (Mm:P3)
[ISUP] adsmGoStandby: (Im1:mId=crnt)
[MTP3] adsmGoStandby: (Mm:mId=crnt)
[RTR] rAddMcastList: (Im1:P4) (Mm:P4)
[RTR] rSetStandbyMap: (Im1:P4) (Mm:P4)
[ISUP] adsmWarmStart: (Im1)
WarmStart for ISUP: Im
[MTP3] adsmWarmStart: (Mm)
WarmStart for MTP3: Mm
C. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (Im2:P4) (I2:P4)
[RTR] rSetActiveMap: (Im2:P4) (I2:P4) (M2:P4) (M3:P4)
[RTR] rSetActiveMap: (Im2:P4) (I2:P4) (M2:P4) (M3:P4)

Figure 21

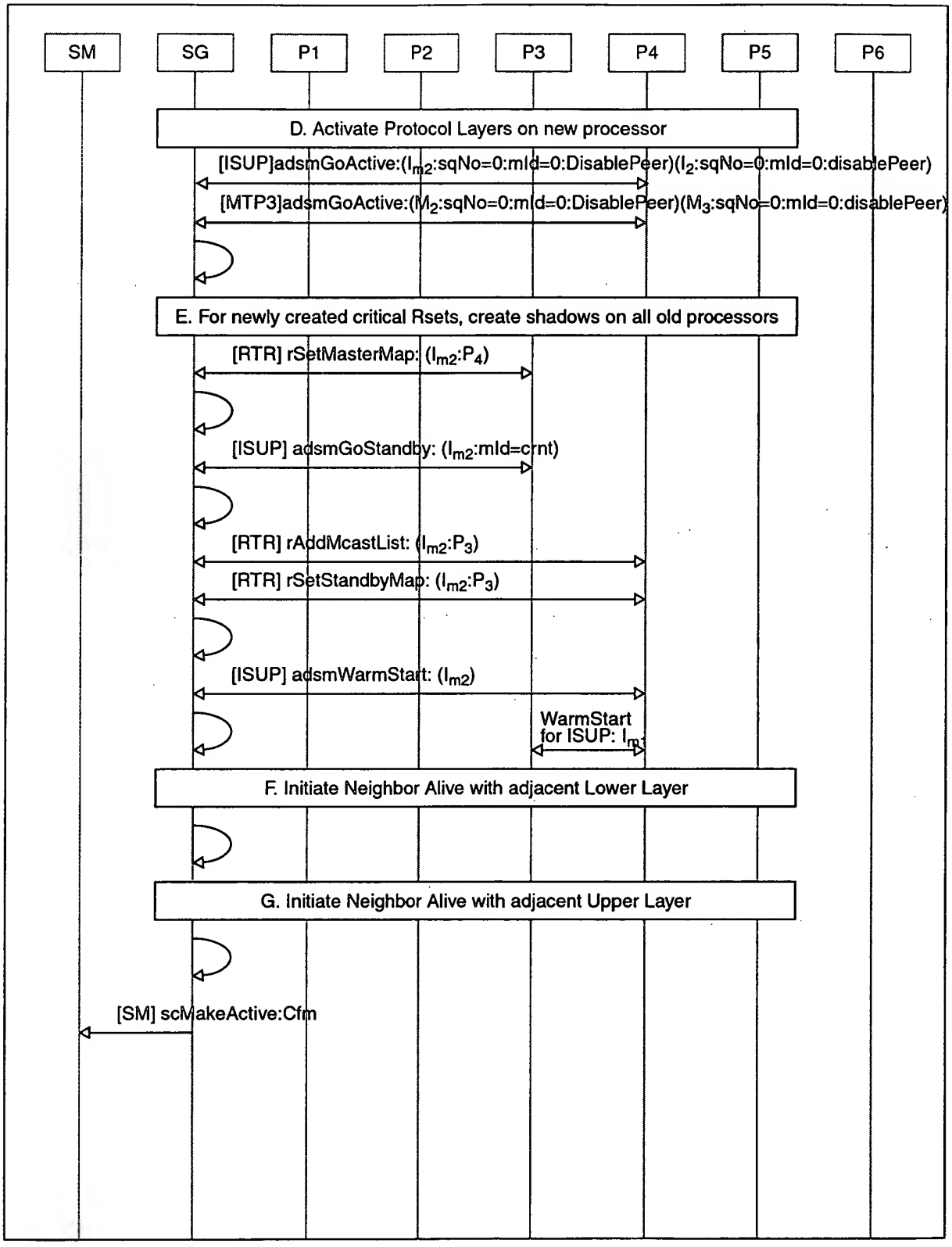
SM
SG
P1
P2
P3
P4
P5
P6
D. Activate Protocol Layers on new processor
[ISUP]adsmGoActive:(Im2:sqNo=0:mId=0:DisablePeer)(I2:sqNo=0:mId=0:disablePeer)
[MTP3]adsmGoActive:(M2:sqNo=0:mId=0:DisablePeer)(M3:sqNo=0:mId=0:disablePeer)
E. For newly created critical Rsets, create shadows on all old processors
[RTR] rSetMasterMap: (Im2:P4)
[ISUP] adsmGoStandby: (Im2:mId=crnt)
[RTR] rAddMcastList: (Im2:P3)
[RTR] rSetStandbyMap: (Im2:P3)
[ISUP] adsmWarmStart: (Im2)
WarmStart for ISUP: Im2
F. Initiate Neighbor Alive with adjacent Lower Layer
G. Initiate Neighbor Alive with adjacent Upper Layer
[SM] scMakeActive:Cfm

Figure 22

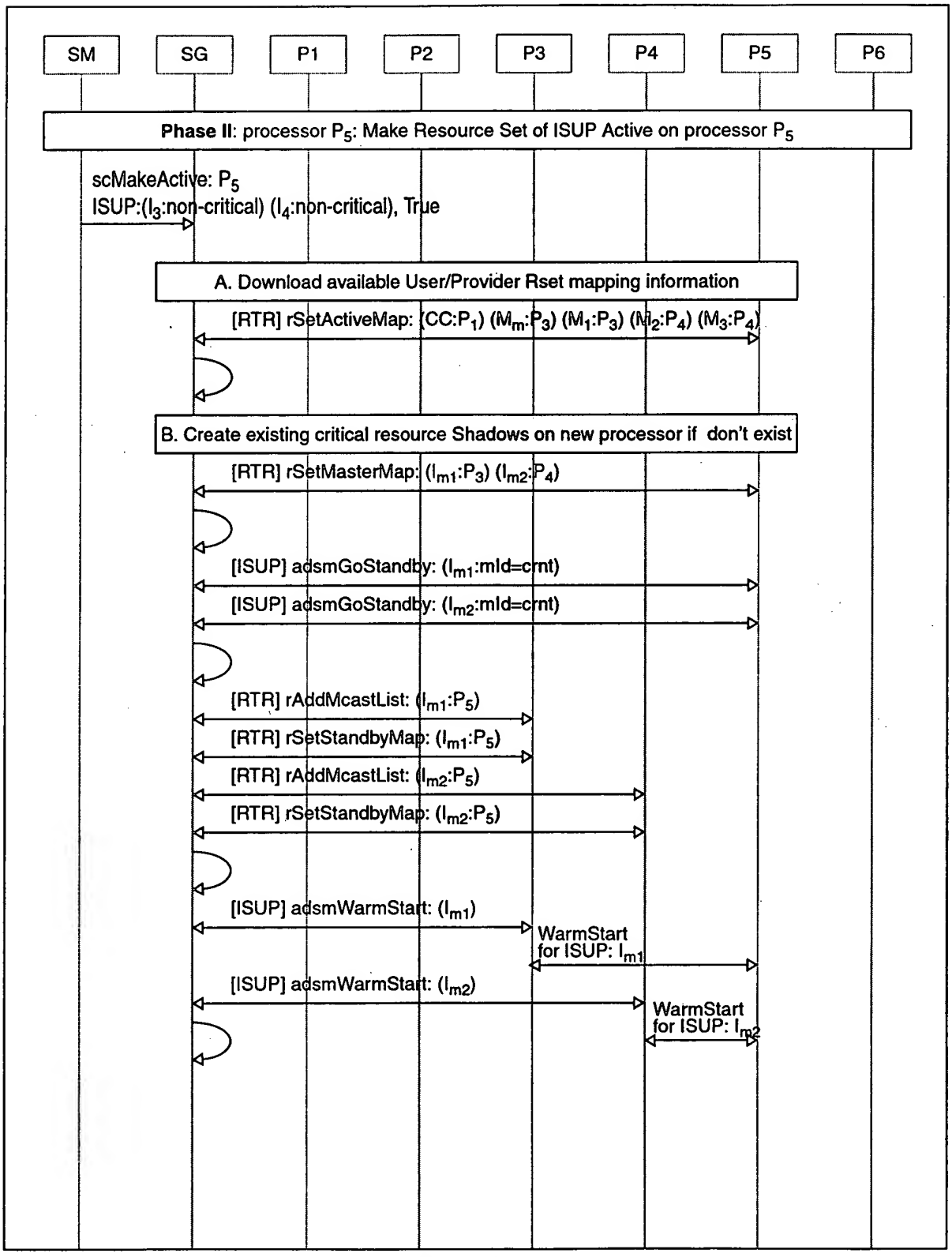
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: processor P5: Make Resource Set of ISUP Active on processor P5
scMakeActive: P5
ISUP:(I3:non-critical) (I4:non-critical), True
A. Download available User/Provider Rset mapping information
[RTR] rSetActiveMap: (CC:P1) (Mm:P3) (M1:P3) (M2:P4) (M3:P4)
B. Create existing critical resource Shadows on new processor if don't exist
[RTR] rSetMasterMap: (Im1:P3) (Im2:P4)
[ISUP] adsmGoStandby: (Im1:mId=crnt)
[ISUP] adsmGoStandby: (Im2:mId=crnt)
[RTR] rAddMcastList: (Im1:P5)
[RTR] rSetStandbyMap: (Im1:P5)
[RTR] rAddMcastList: (Im2:P5)
[RTR] rSetStandbyMap: (Im2:P5)
[ISUP] adsmWarmStart: (Im1)
WarmStart for ISUP: Im1
[ISUP] adsmWarmStart: (Im2)
WarmStart for ISUP: Im2

Figure 23

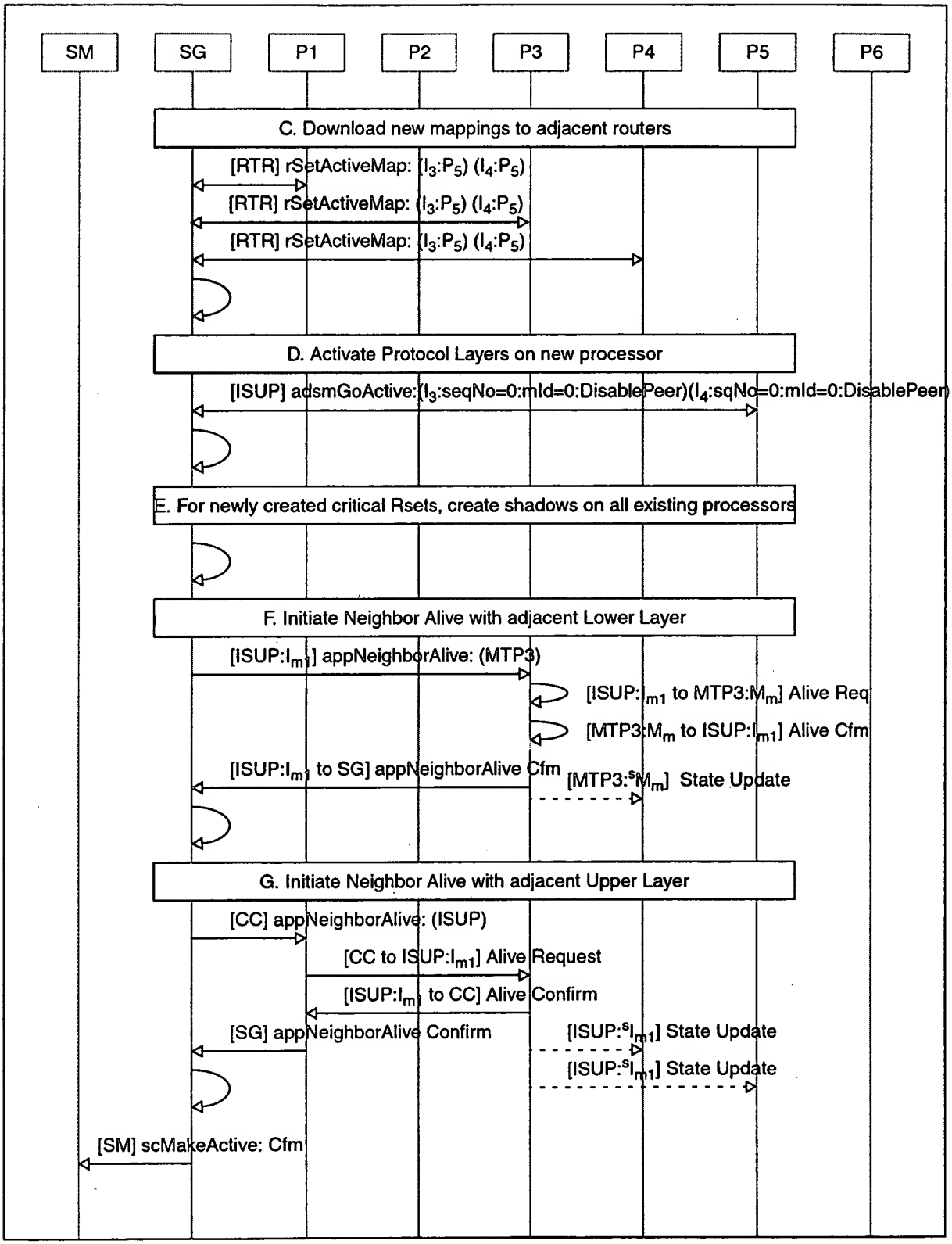
SM
SG
P1
P2
P3
P4
P5
P6
C. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (I3:P5) (I4:P5)
[RTR] rSetActiveMap: (I3:P5) (I4:P5)
[RTR] rSetActiveMap: (I3:P5) (I4:P5)
D. Activate Protocol Layers on new processor
[ISUP] adsmGoActive:(I3:seqNo=0:mId=0:DisablePeer)(I4:sqNo=0:mId=0:DisablePeer)
E. For newly created critical Rsets, create shadows on all existing processors
F. Initiate Neighbor Alive with adjacent Lower Layer
[ISUP:Im1] appNeighborAlive: (MTP3)
[ISUP:Im1 to MTP3:Mm] Alive Req
[MTP3:Mm to ISUP:Im1] Alive Cfm
[ISUP:Im1 to SG] appNeighborAlive Cfm
[MTP3:sMm] State Update
G. Initiate Neighbor Alive with adjacent Upper Layer
[CC] appNeighborAlive: (ISUP)
[CC to ISUP:Im1] Alive Request
[ISUP:Im1 to CC] Alive Confirm
[SG] appNeighborAlive Confirm
[ISUP:sIm1] State Update
[ISUP:sIm1] State Update
[SM] scMakeActive: Cfm

Figure 24

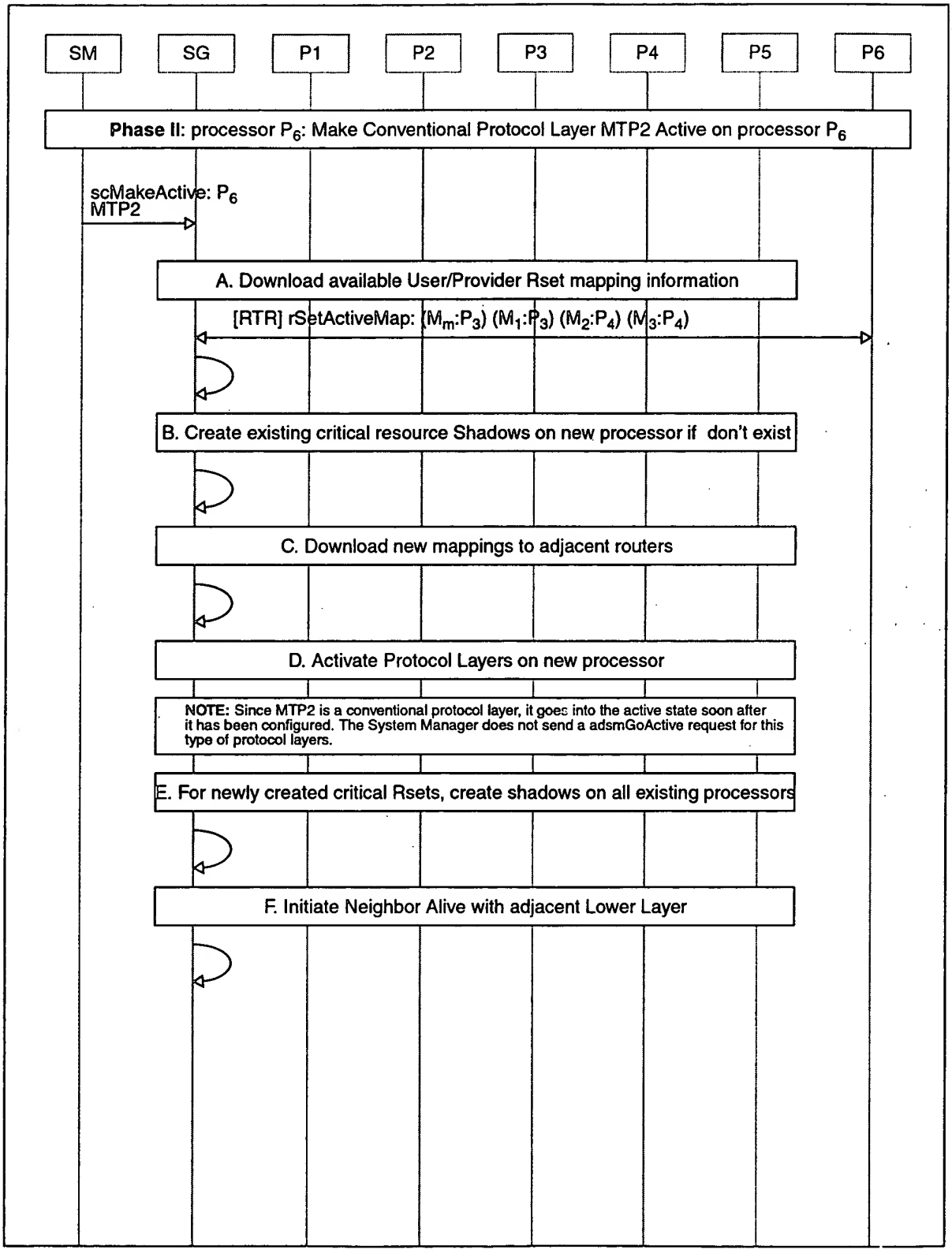
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: processor P_6: Make Conventional Protocol Layer MTP2 Active on processor P_6
scMakeActive: P_6
MTP2
A. Download available User/Provider Rset mapping information
[RTR] rSetActiveMap: (M_m:P_3) (M_1:P_3) (M_2:P_4) (M_3:P_4)
B. Create existing critical resource Shadows on new processor if don't exist
C. Download new mappings to adjacent routers
D. Activate Protocol Layers on new processor
NOTE: Since MTP2 is a conventional protocol layer, it goes into the active state soon after it has been configured. The System Manager does not send a adsmGoActive request for this type of protocol layers.
E. For newly created critical Rsets, create shadows on all existing processors
F. Initiate Neighbor Alive with adjacent Lower Layer

Figure 25

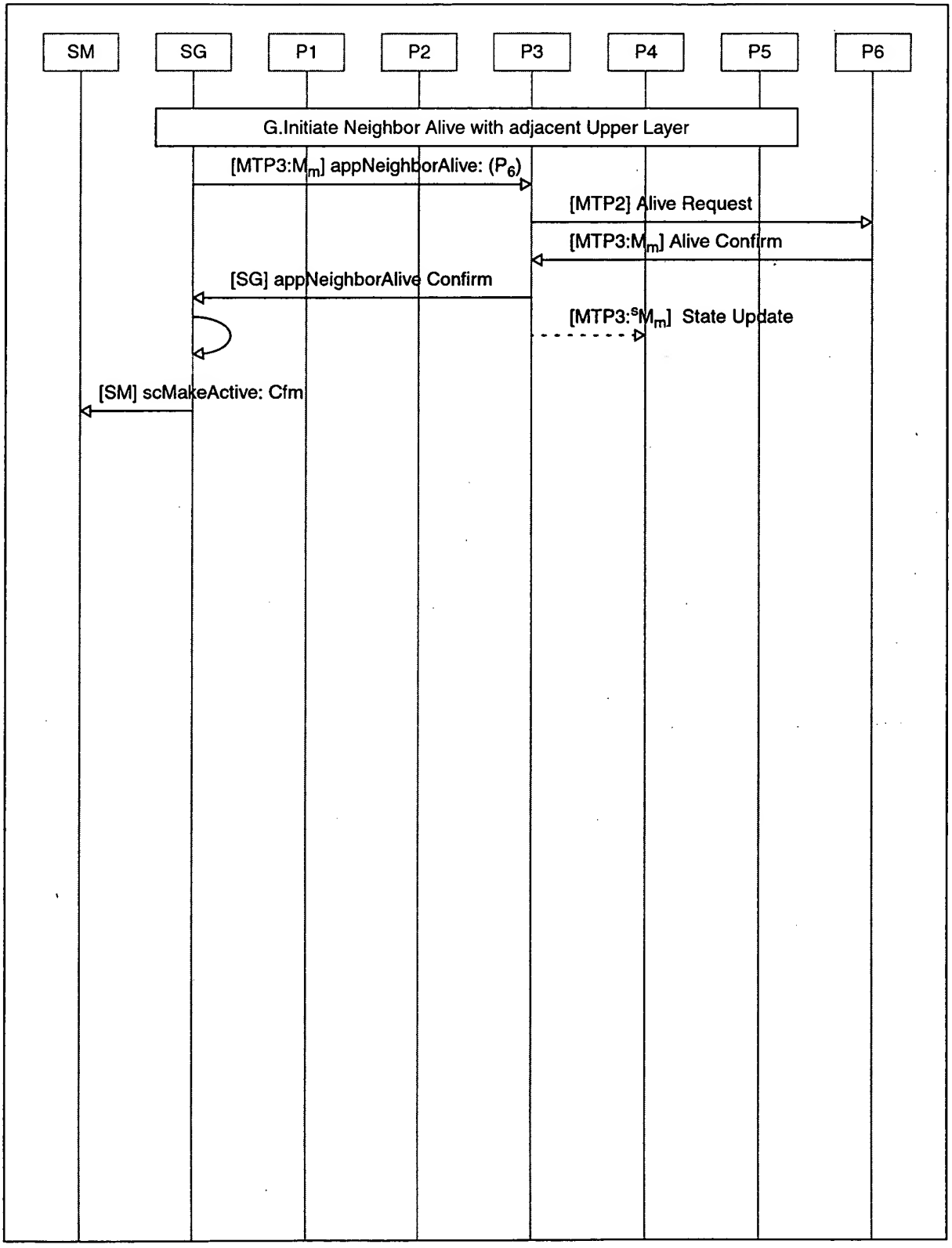
SM
SG
P1
P2
P3
P4
P5
P6
G.Initiate Neighbor Alive with adjacent Upper Layer
[MTP3:M_m] appNeighborAlive: (P_6)
[MTP2] Alive Request
[MTP3:M_m] Alive Confirm
[SG] appNeighborAlive Confirm
[MTP3:^sM_m] State Update
[SM] scMakeActive: Cfm

Figure 26

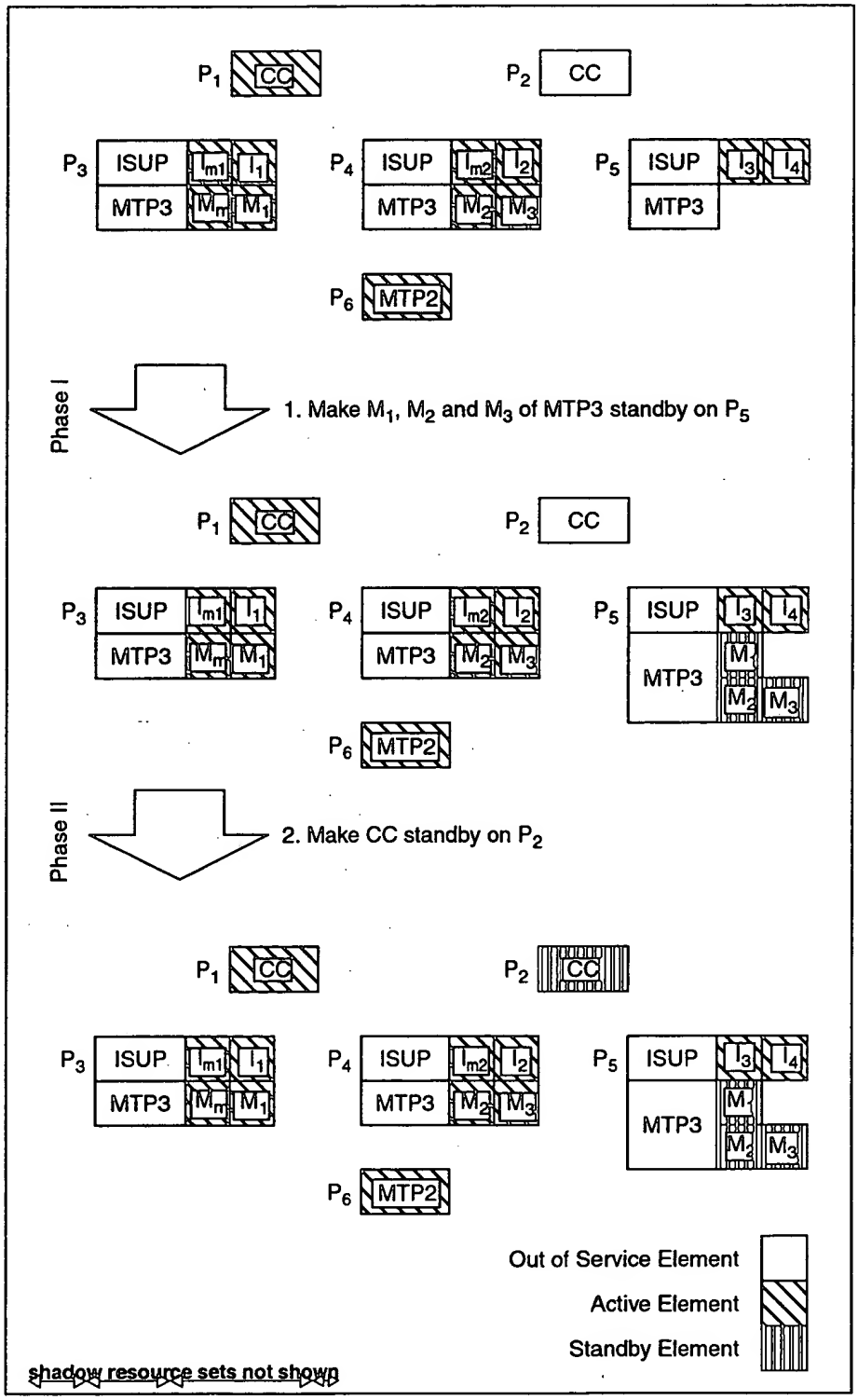
P1 CC
P2 CC
P3 ISUP Im1 I1 MTP3 Mm M1
P4 ISUP Im2 I2 MTP3 M2 M3
P5 ISUP I3 I4 MTP3
P6 MTP2
Phase I
1. Make M1, M2 and M3 of MTP3 standby on P5
P1 CC
P2 CC
P3 ISUP Im1 I1 MTP3 Mm M1
P4 ISUP Im2 I2 MTP3 M2 M3
P5 ISUP I3 I4 MTP3 M1 M2 M3
P6 MTP2
Phase II
2. Make CC standby on P2
P1 CC
P2 CC
P3 ISUP Im1 I1 MTP3 Mm M1
P4 ISUP Im2 I2 MTP3 M2 M3
P5 ISUP I3 I4 MTP3 M1 M2 M3
P6 MTP2
Out of Service Element
Active Element
Standby Element
shadow resource sets not shown

Figure 27

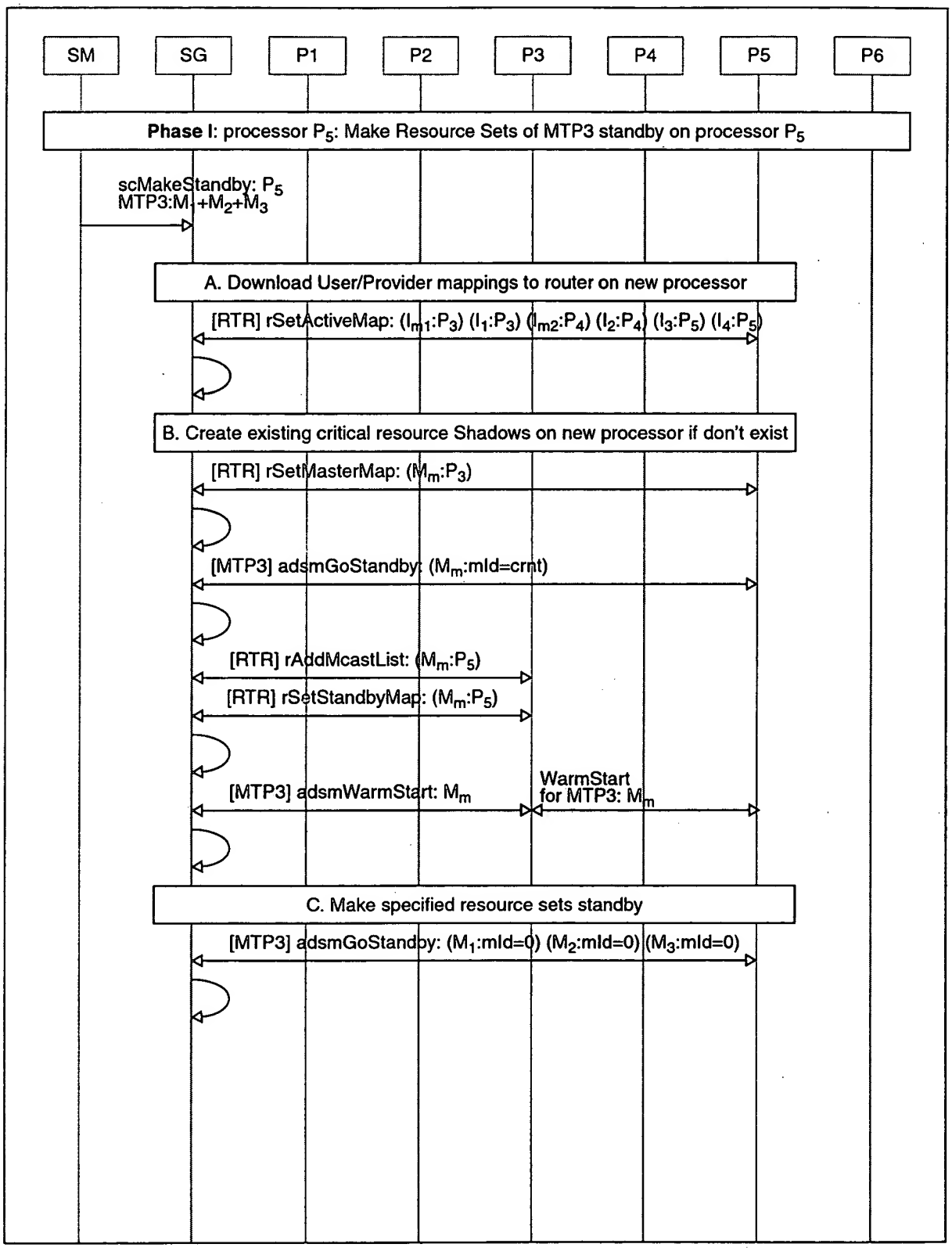
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: processor P5: Make Resource Sets of MTP3 standby on processor P5
scMakeStandby: P5
MTP3:M1+M2+M3
A. Download User/Provider mappings to router on new processor
[RTR] rSetActiveMap: (Im1:P3) (I1:P3) (Im2:P4) (I2:P4) (I3:P5) (I4:P5)
B. Create existing critical resource Shadows on new processor if don't exist
[RTR] rSetMasterMap: (Mm:P3)
[MTP3] adsmGoStandby: (Mm:mId=crnt)
[RTR] rAddMcastList: (Mm:P5)
[RTR] rSetStandbyMap: (Mm:P5)
[MTP3] adsmWarmStart: Mm
WarmStart
for MTP3: Mm
C. Make specified resource sets standby
[MTP3] adsmGoStandby: (M1:mId=0) (M2:mId=0) (M3:mId=0)

Figure 28

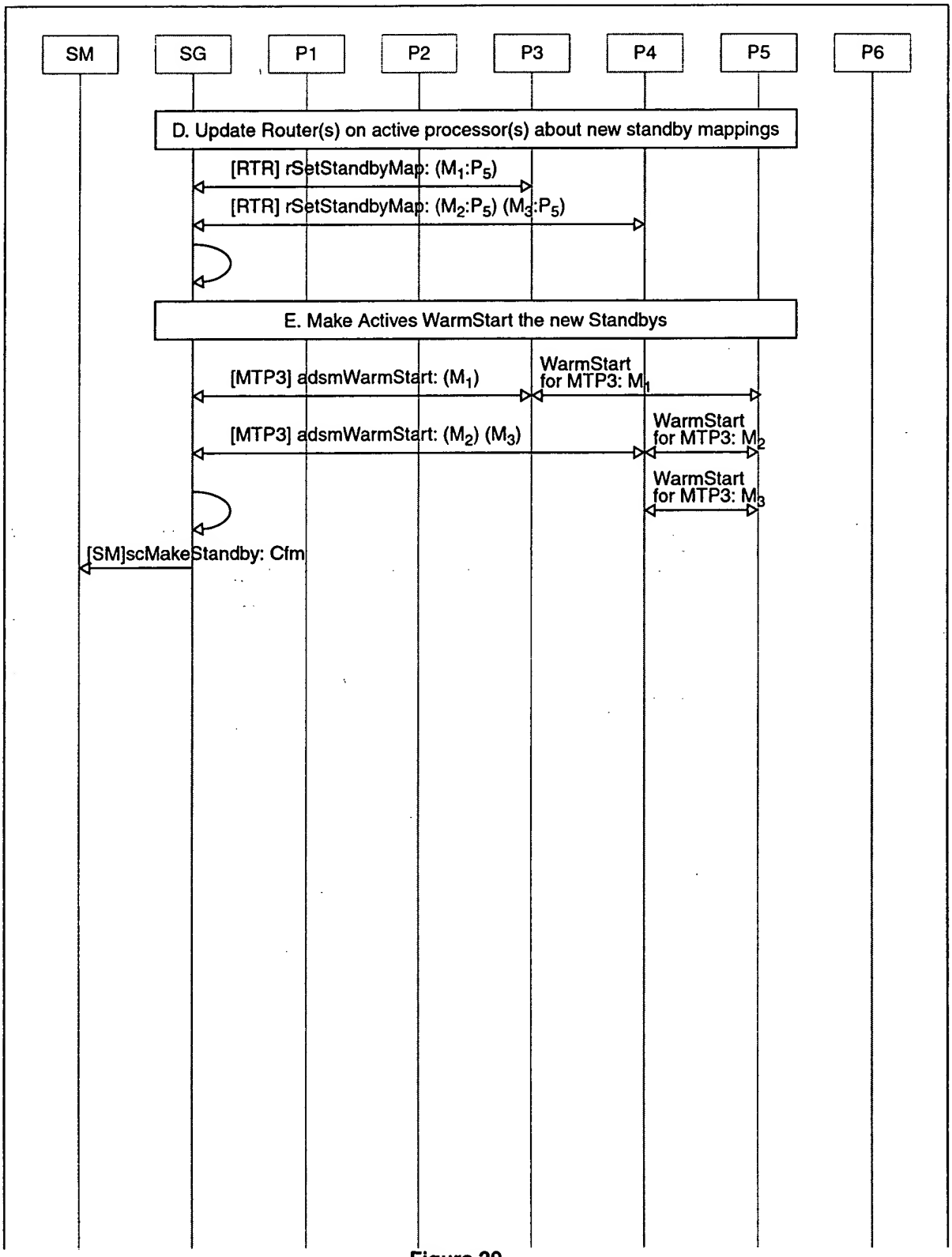
SM
SG
P1
P2
P3
P4
P5
P6
D. Update Router(s) on active processor(s) about new standby mappings
[RTR] rSetStandbyMap: (M1:P5)
[RTR] rSetStandbyMap: (M2:P5) (M3:P5)
E. Make Actives WarmStart the new Standbys
[MTP3] adsmWarmStart: (M1)
WarmStart for MTP3: M1
[MTP3] adsmWarmStart: (M2) (M3)
WarmStart for MTP3: M2
WarmStart for MTP3: M3
[SM]scMakeStandby: Cfm

Figure 29

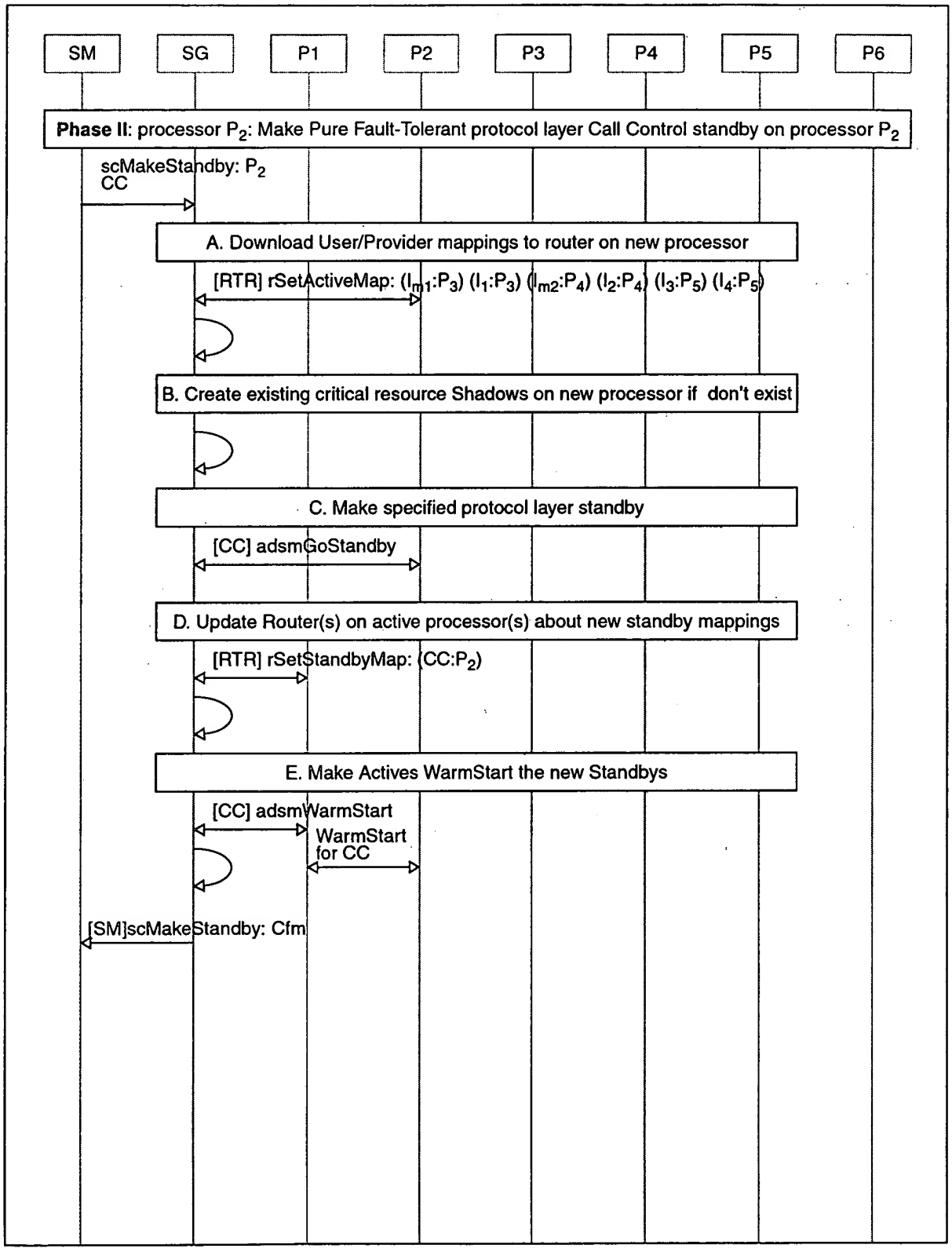
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: processor P2: Make Pure Fault-Tolerant protocol layer Call Control standby on processor P2
scMakeStandby: P2 CC
A. Download User/Provider mappings to router on new processor
[RTR] rSetActiveMap: (Im1:P3) (I1:P3) (Im2:P4) (I2:P4) (I3:P5) (I4:P5)
B. Create existing critical resource Shadows on new processor if don't exist
C. Make specified protocol layer standby
[CC] adsmGoStandby
D. Update Router(s) on active processor(s) about new standby mappings
[RTR] rSetStandbyMap: (CC:P2)
E. Make Actives WarmStart the new Standbys
[CC] adsmWarmStart
WarmStart for CC
[SM]scMakeStandby: Cfm

Figure 30

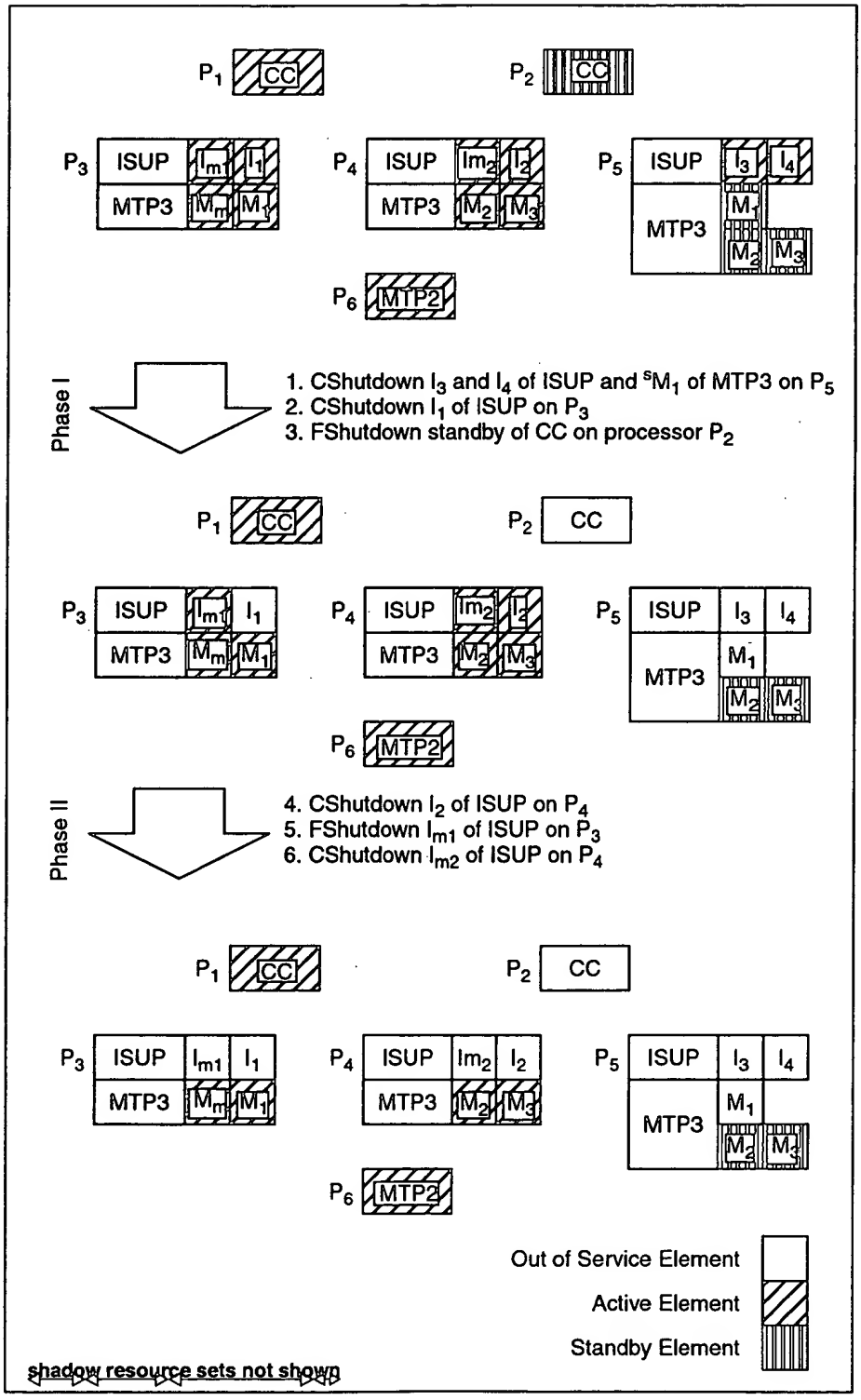
P1 CC
P2 CC
P3 ISUP Im1 I1
MTP3 Mm M1
P4 ISUP Im2 I2
MTP3 M2 M3
P5 ISUP I3 I4
MTP3 M1 M2 M3
P6 MTP2
Phase I
1. CShutdown I3 and I4 of ISUP and sM1 of MTP3 on P5
2. CShutdown I1 of ISUP on P3
3. FShutdown standby of CC on processor P2
P1 CC
P2 CC
P3 ISUP Im1 I1
MTP3 Mm M1
P4 ISUP Im2 I2
MTP3 M2 M3
P5 ISUP I3 I4
MTP3 M1 M2 M3
P6 MTP2
Phase II
4. CShutdown I2 of ISUP on P4
5. FShutdown Im1 of ISUP on P3
6. CShutdown Im2 of ISUP on P4
P1 CC
P2 CC
P3 ISUP Im1 I1
MTP3 Mm M1
P4 ISUP Im2 I2
MTP3 M2 M3
P5 ISUP I3 I4
MTP3 M1 M2 M3
P6 MTP2
Out of Service Element
Active Element
Standby Element
shadow resource sets not shown

Figure 31

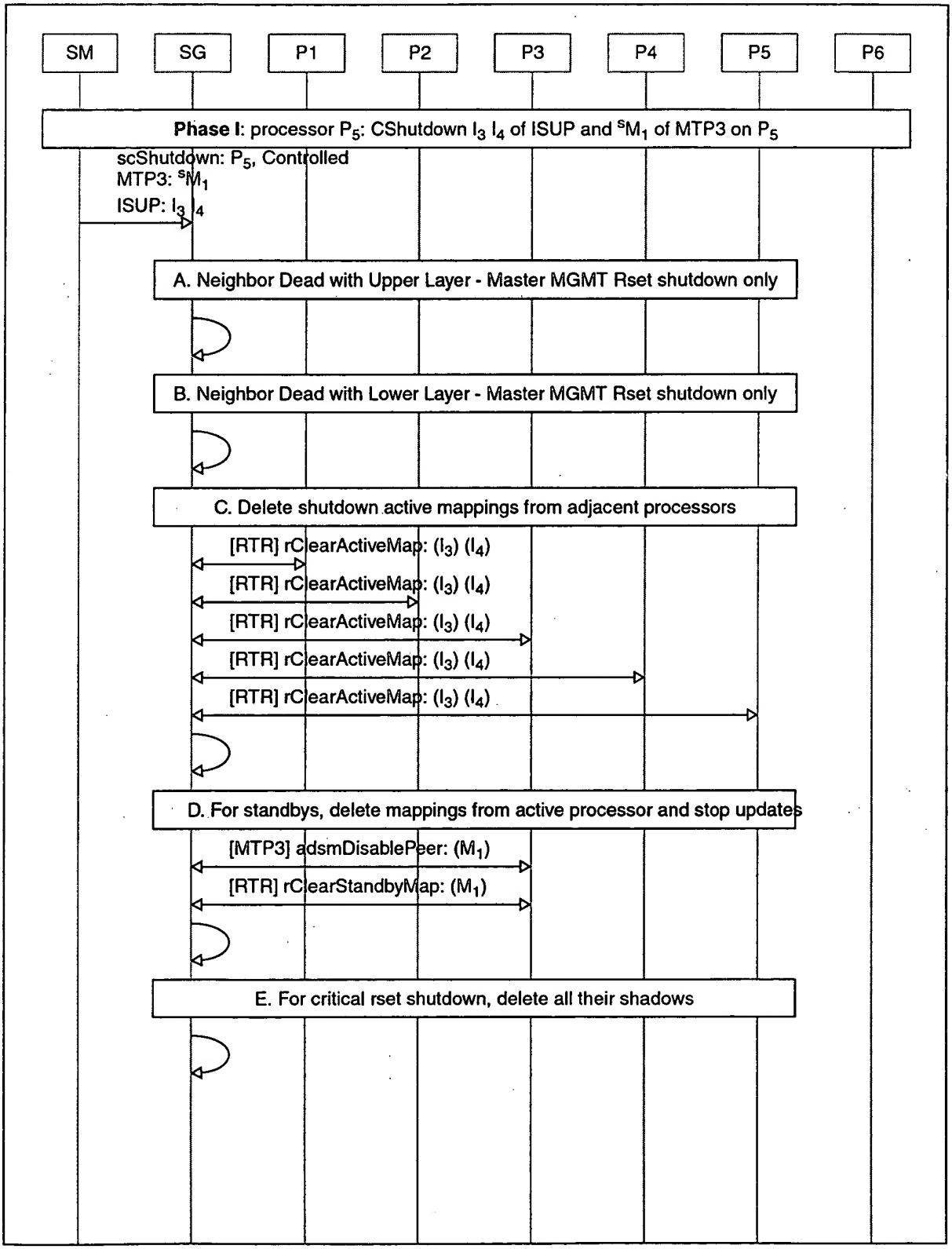
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: processor P5: CShutdown I3 I4 of ISUP and SM1 of MTP3 on P5
scShutdown: P5, Controlled
MTP3: SM1
ISUP: I3 I4
A. Neighbor Dead with Upper Layer - Master MGMT Rset shutdown only
B. Neighbor Dead with Lower Layer - Master MGMT Rset shutdown only
C. Delete shutdown active mappings from adjacent processors
[RTR] rClearActiveMap: (I3) (I4)
[RTR] rClearActiveMap: (I3) (I4)
[RTR] rClearActiveMap: (I3) (I4)
[RTR] rClearActiveMap: (I3) (I4)
[RTR] rClearActiveMap: (I3) (I4)
D. For standbys, delete mappings from active processor and stop updates
[MTP3] adsmDisablePeer: (M1)
[RTR] rClearStandbyMap: (M1)
E. For critical rset shutdown, delete all their shadows

Figure 32

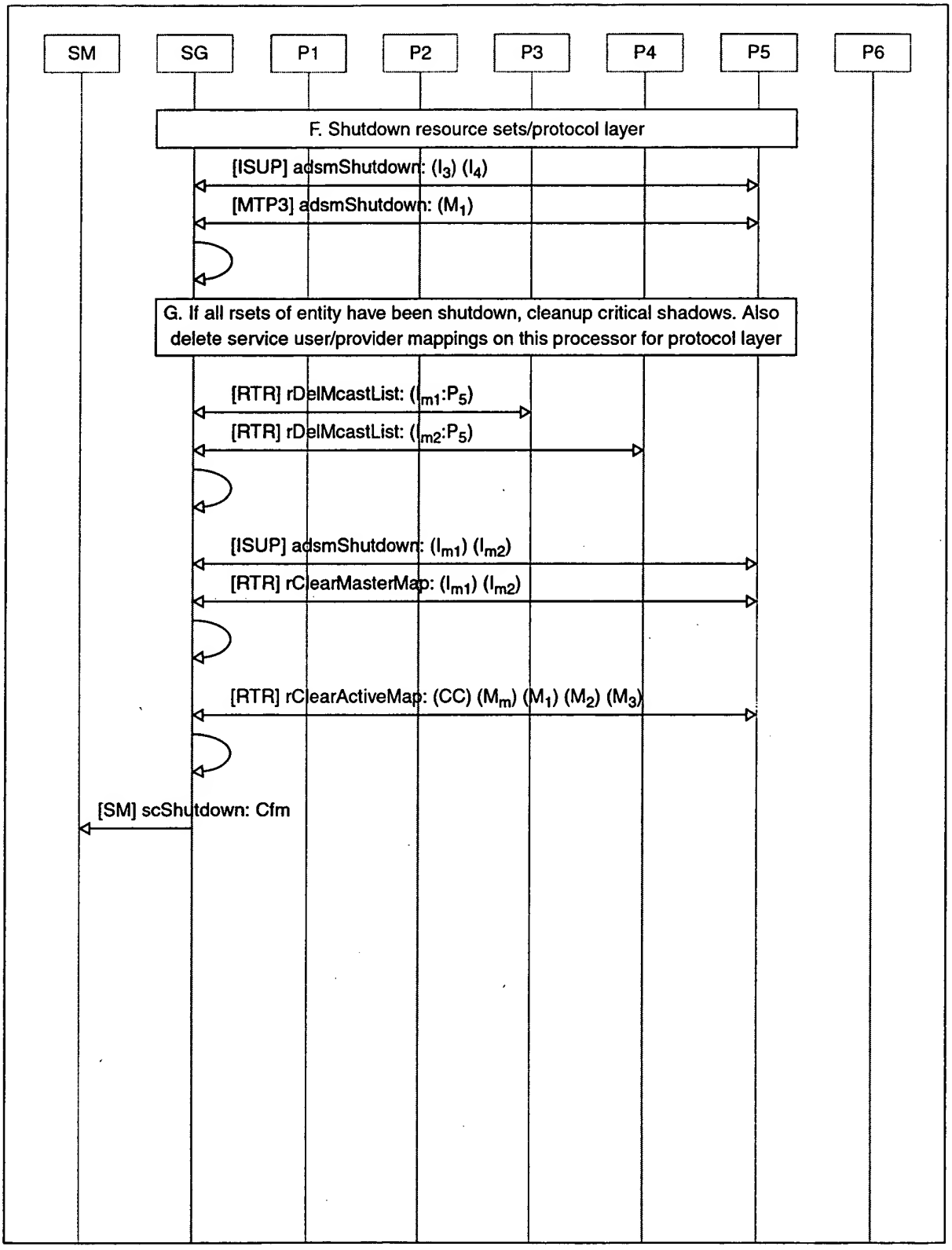
SM
SG
P1
P2
P3
P4
P5
P6
F. Shutdown resource sets/protocol layer
[ISUP] adsmShutdown: (I3) (I4)
[MTP3] adsmShutdown: (M1)
G. If all rsets of entity have been shutdown, cleanup critical shadows. Also delete service user/provider mappings on this processor for protocol layer
[RTR] rDelMcastList: (Im1:P5)
[RTR] rDelMcastList: (Im2:P5)
[ISUP] adsmShutdown: (Im1) (Im2)
[RTR] rClearMasterMap: (Im1) (Im2)
[RTR] rClearActiveMap: (CC) (Mm) (M1) (M2) (M3)
[SM] scShutdown: Cfm

Figure 33

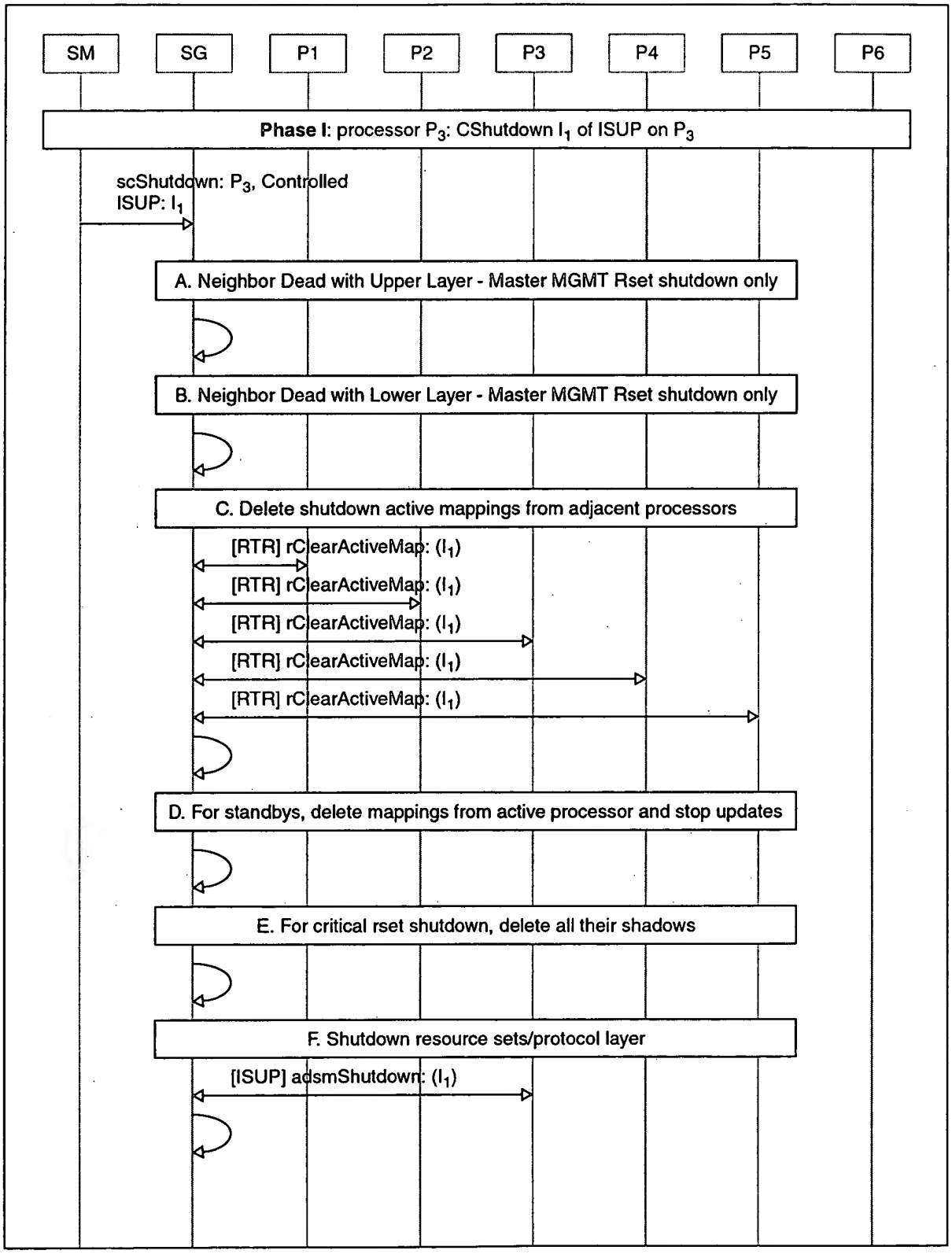
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: processor P3: CShutdown I1 of ISUP on P3
scShutdown: P3, Controlled
ISUP: I1
A. Neighbor Dead with Upper Layer - Master MGMT Rset shutdown only
B. Neighbor Dead with Lower Layer - Master MGMT Rset shutdown only
C. Delete shutdown active mappings from adjacent processors
[RTR] rClearActiveMap: (I1)
[RTR] rClearActiveMap: (I1)
[RTR] rClearActiveMap: (I1)
[RTR] rClearActiveMap: (I1)
[RTR] rClearActiveMap: (I1)
D. For standbys, delete mappings from active processor and stop updates
E. For critical rset shutdown, delete all their shadows
F. Shutdown resource sets/protocol layer
[ISUP] adsmShutdown: (I1)

Figure 34

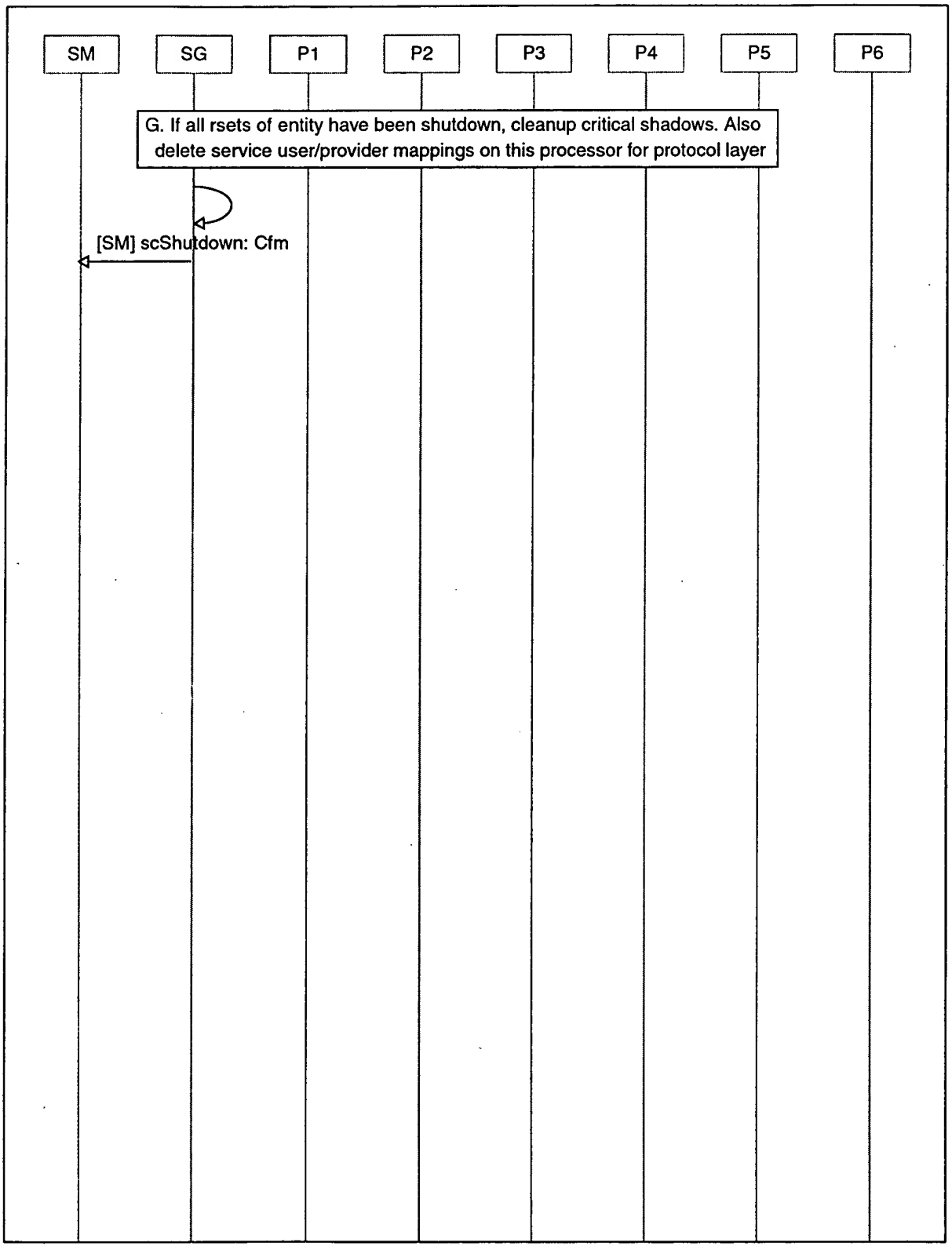
SM
SG
P1
P2
P3
P4
P5
P6
G. If all rsets of entity have been shutdown, cleanup critical shadows. Also delete service user/provider mappings on this processor for protocol layer
[SM] scShutdown: Cfm

Figure 35

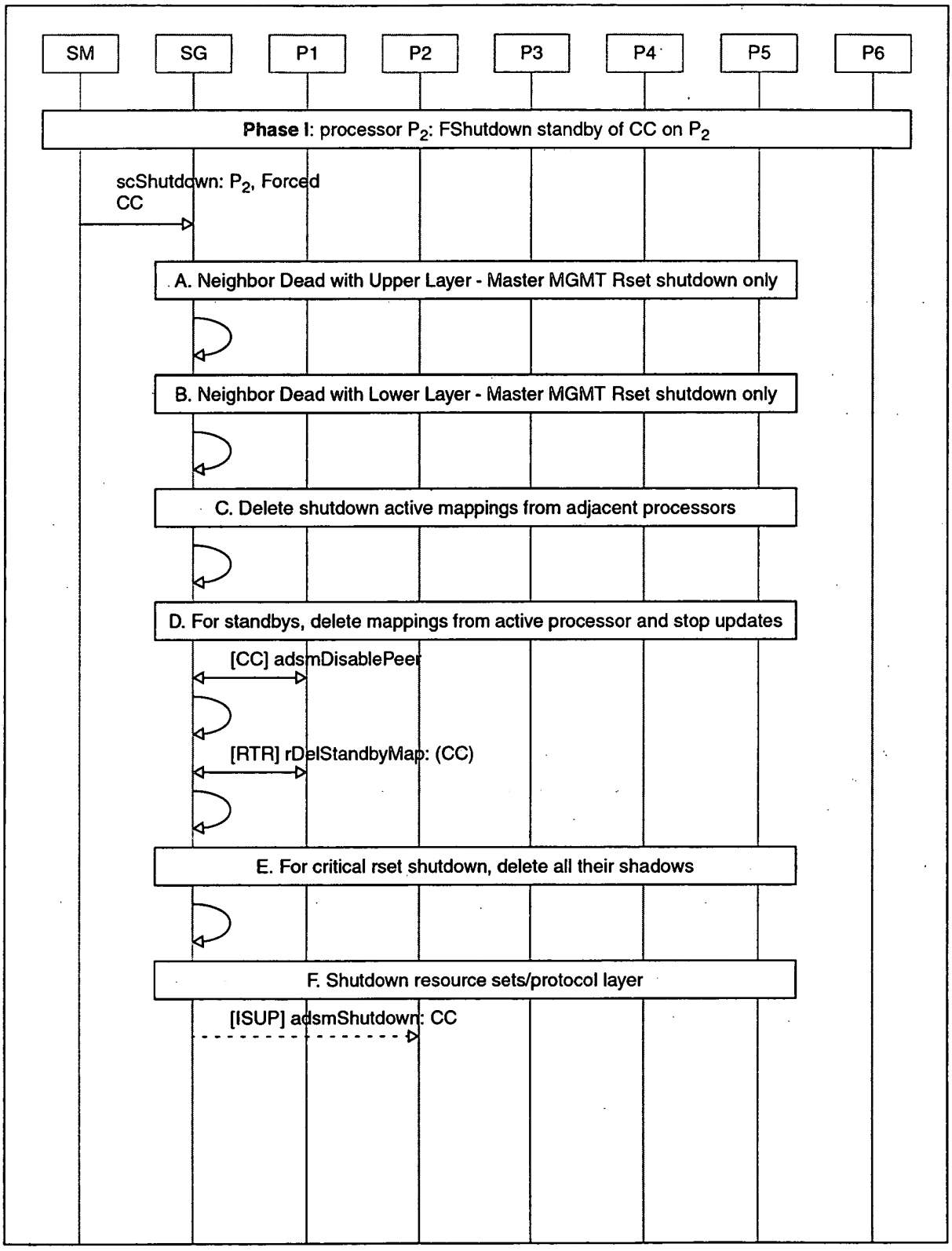
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: processor P2: FShutdown standby of CC on P2
scShutdown: P2, Forced CC
A. Neighbor Dead with Upper Layer - Master MGMT Rset shutdown only
B. Neighbor Dead with Lower Layer - Master MGMT Rset shutdown only
C. Delete shutdown active mappings from adjacent processors
D. For standbys, delete mappings from active processor and stop updates
[CC] adsmDisablePeer
[RTR] rDelStandbyMap: (CC)
E. For critical rset shutdown, delete all their shadows
F. Shutdown resource sets/protocol layer
[ISUP] adsmShutdown: CC

Figure 36

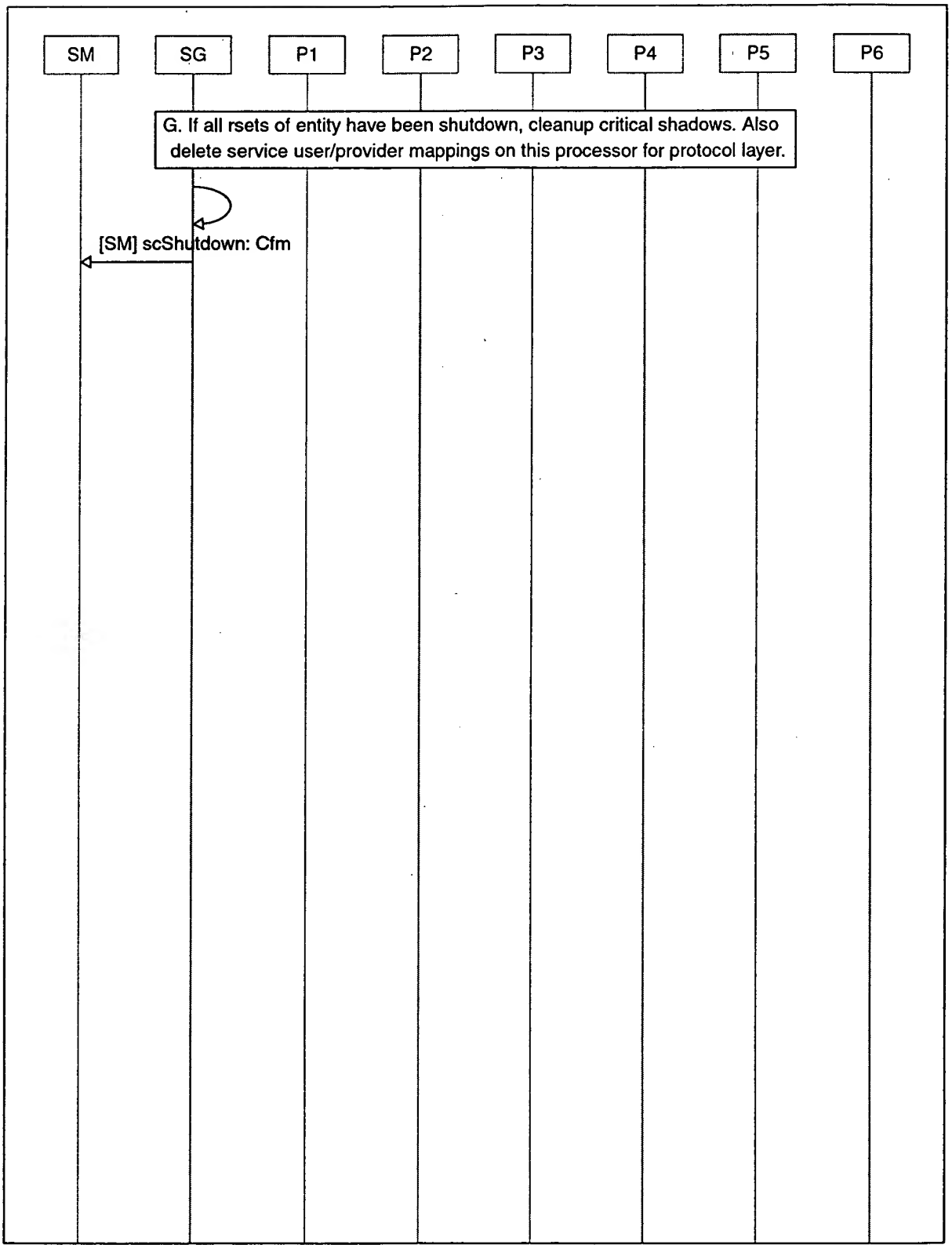
SM
SG
P1
P2
P3
P4
P5
P6
G. If all rsets of entity have been shutdown, cleanup critical shadows. Also delete service user/provider mappings on this processor for protocol layer.
[SM] scShutdown: Cfm

Figure 37

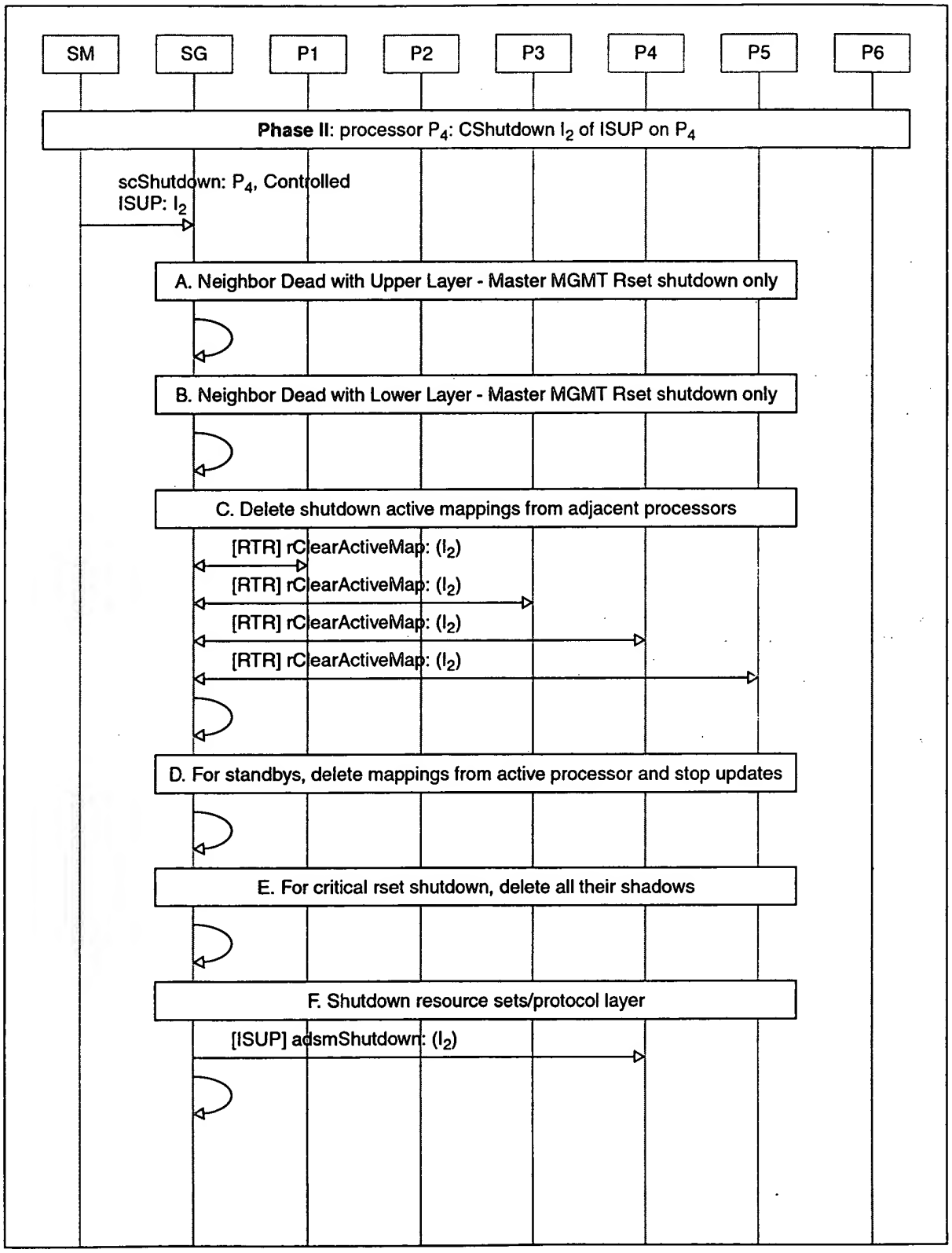
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: processor P4: CShutdown I2 of ISUP on P4
scShutdown: P4, Controlled
ISUP: I2
A. Neighbor Dead with Upper Layer - Master MGMT Rset shutdown only
B. Neighbor Dead with Lower Layer - Master MGMT Rset shutdown only
C. Delete shutdown active mappings from adjacent processors
[RTR] rClearActiveMap: (I2)
[RTR] rClearActiveMap: (I2)
[RTR] rClearActiveMap: (I2)
[RTR] rClearActiveMap: (I2)
D. For standbys, delete mappings from active processor and stop updates
E. For critical rset shutdown, delete all their shadows
F. Shutdown resource sets/protocol layer
[ISUP] adsmShutdown: (I2)

Figure 38

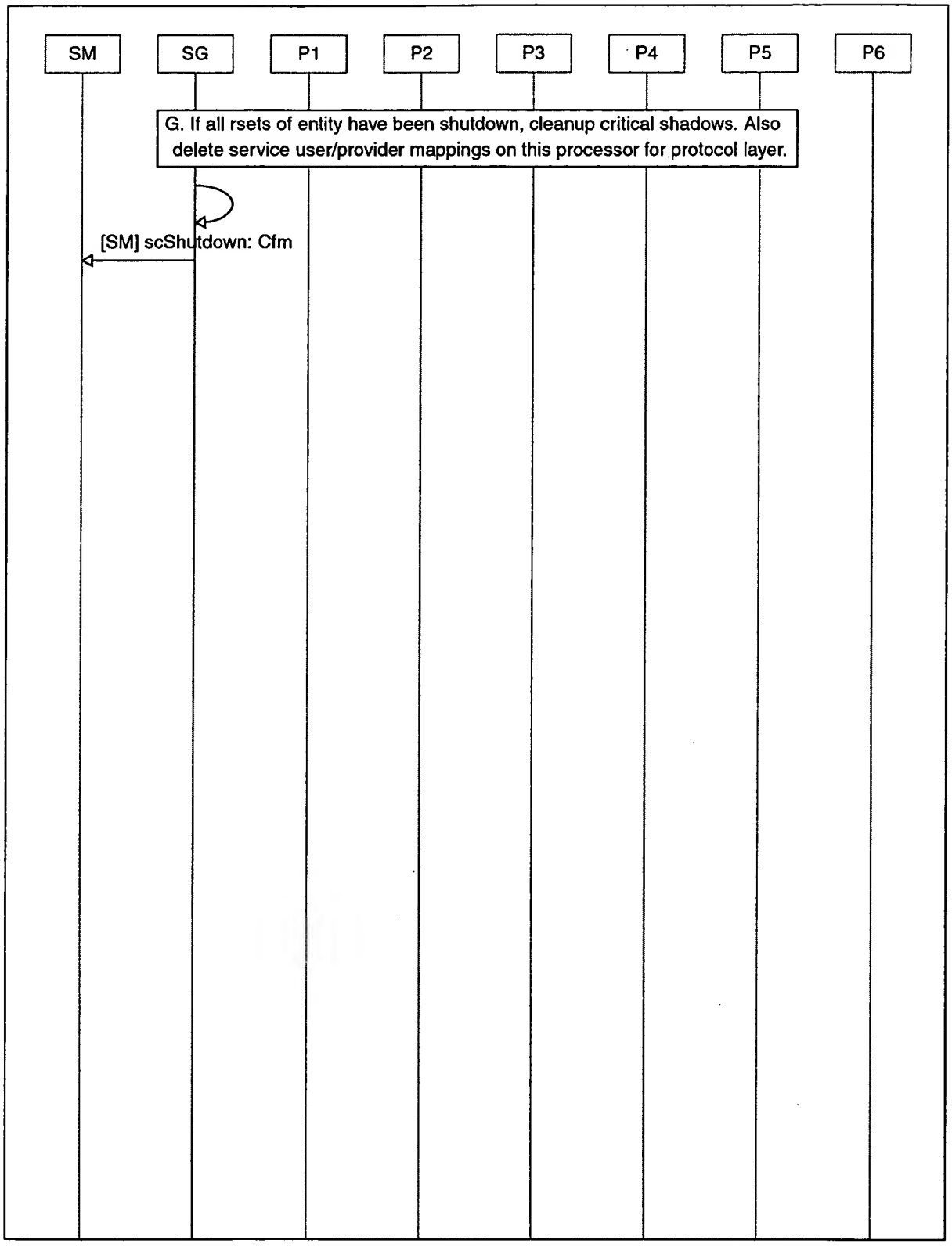
SM
SG
P1
P2
P3
P4
P5
P6
G. If all rsets of entity have been shutdown, cleanup critical shadows. Also delete service user/provider mappings on this processor for protocol layer.
[SM] scShutdown: Cfm

Figure 39

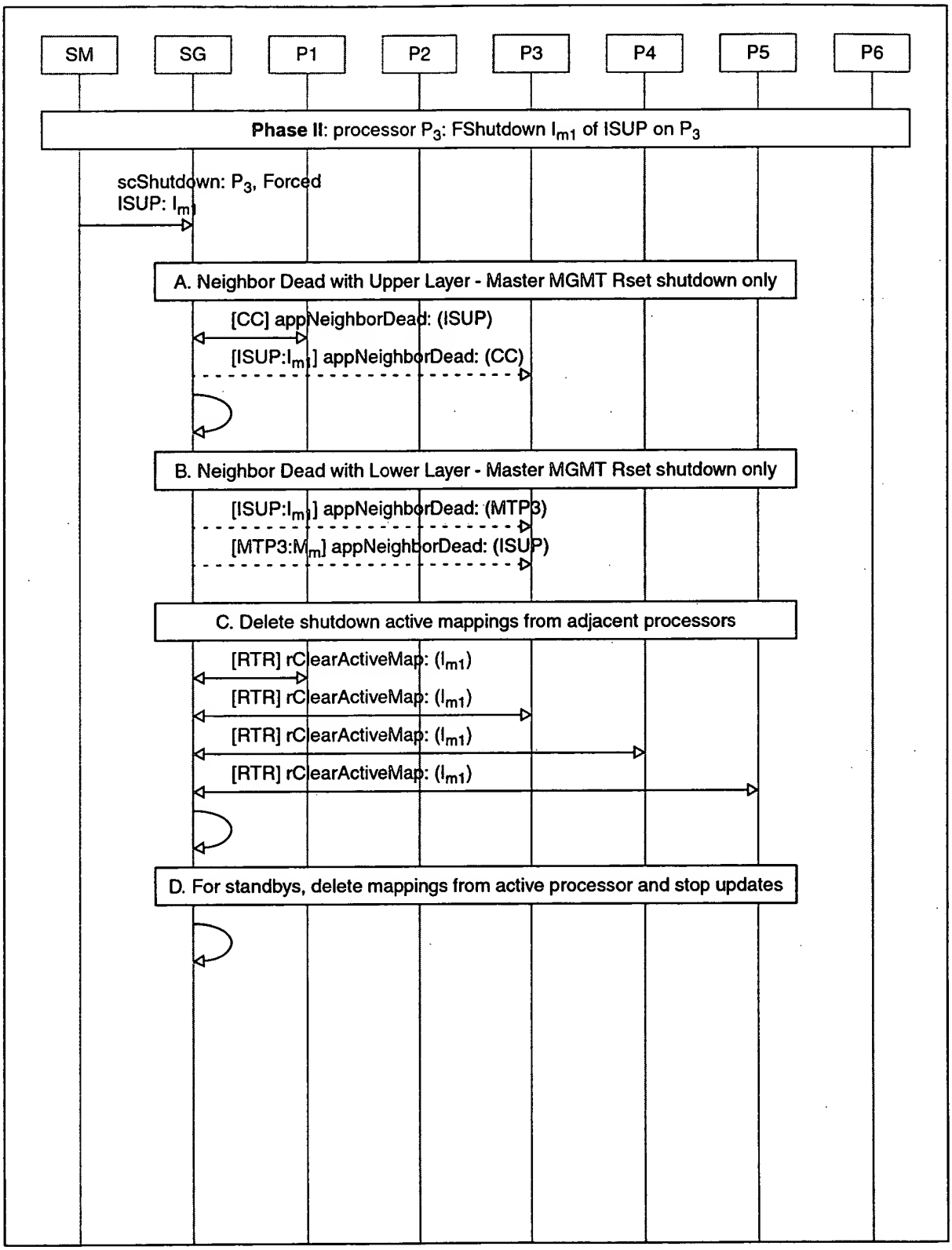
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: processor P_3: FShutdown I_{m1} of ISUP on P_3
scShutdown: P_3, Forced
ISUP: I_{m1}
A. Neighbor Dead with Upper Layer - Master MGMT Rset shutdown only
[CC] appNeighborDead: (ISUP)
[ISUP:I_{m1}] appNeighborDead: (CC)
B. Neighbor Dead with Lower Layer - Master MGMT Rset shutdown only
[ISUP:I_{m1}] appNeighborDead: (MTP3)
[MTP3:M_m] appNeighborDead: (ISUP)
C. Delete shutdown active mappings from adjacent processors
[RTR] rClearActiveMap: (I_{m1})
[RTR] rClearActiveMap: (I_{m1})
[RTR] rClearActiveMap: (I_{m1})
[RTR] rClearActiveMap: (I_{m1})
D. For standbys, delete mappings from active processor and stop updates

Figure 40

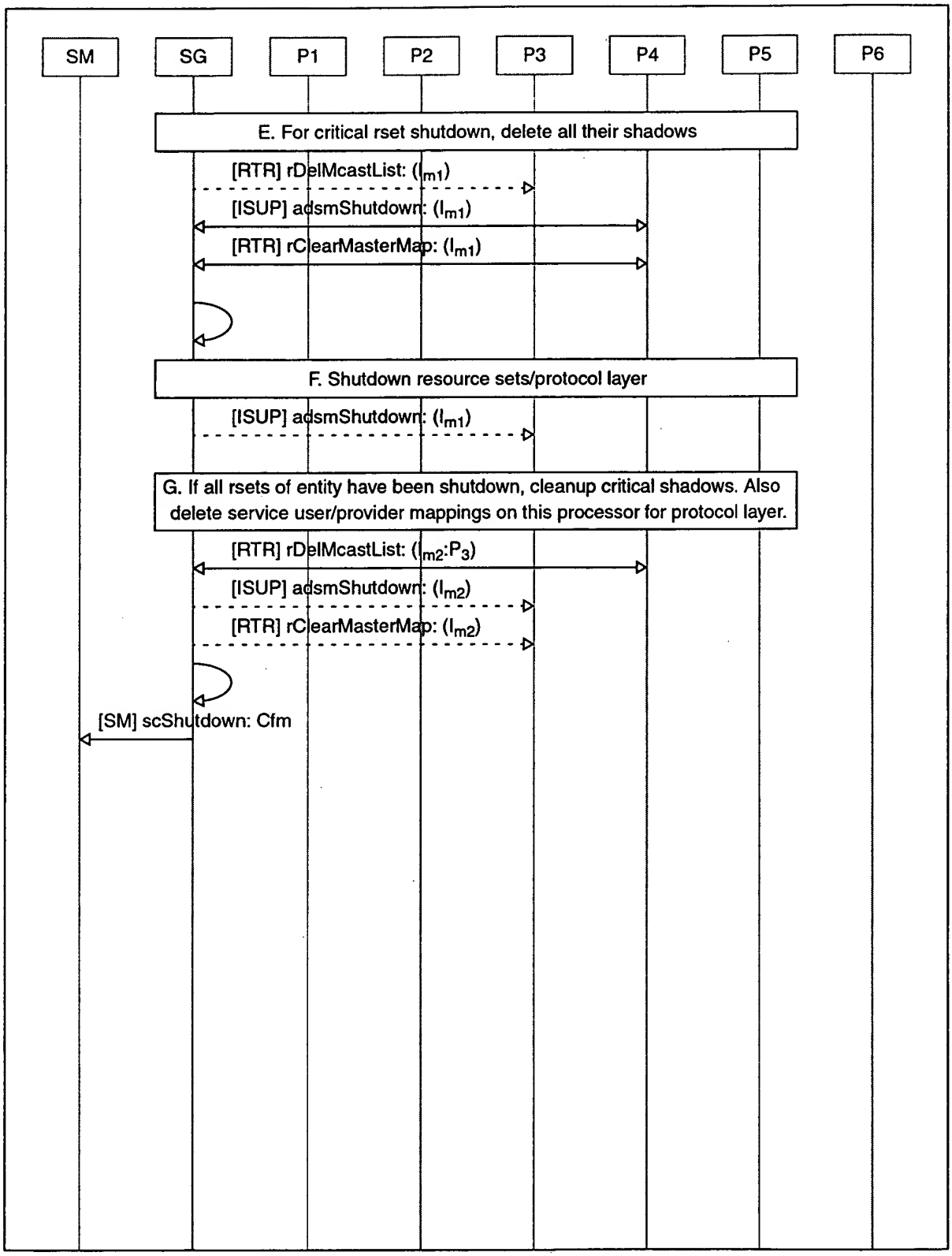
SM
SG
P1
P2
P3
P4
P5
P6
E. For critical rset shutdown, delete all their shadows
[RTR] rDelMcastList: (l_{m1})
[ISUP] adsmShutdown: (l_{m1})
[RTR] rClearMasterMap: (l_{m1})
F. Shutdown resource sets/protocol layer
[ISUP] adsmShutdown: (l_{m1})
G. If all rsets of entity have been shutdown, cleanup critical shadows. Also delete service user/provider mappings on this processor for protocol layer.
[RTR] rDelMcastList: (l_{m2}:P_3)
[ISUP] adsmShutdown: (l_{m2})
[RTR] rClearMasterMap: (l_{m2})
[SM] scShutdown: Cfm

Figure 41

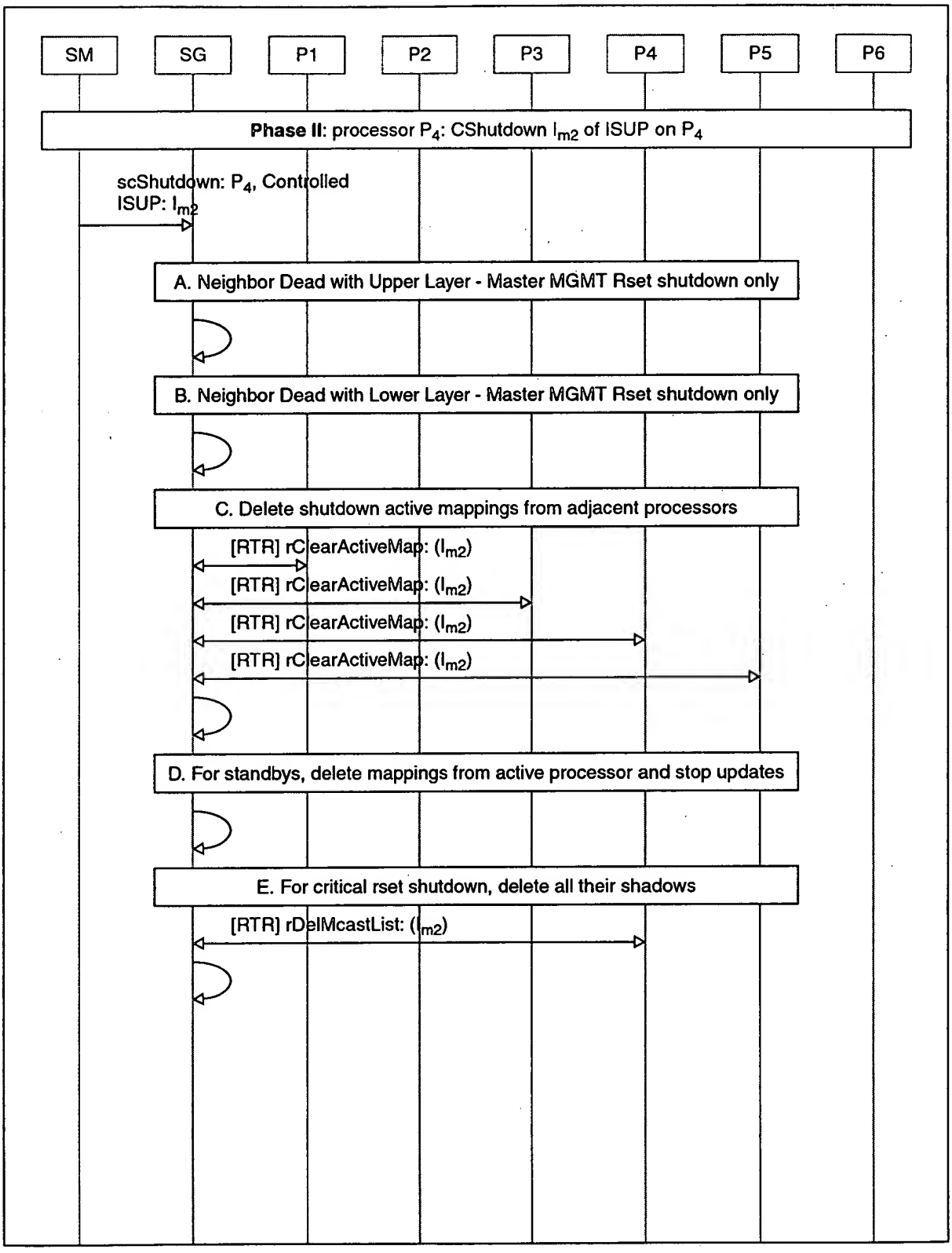
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: processor P4: CShutdown Im2 of ISUP on P4
scShutdown: P4, Controlled
ISUP: Im2
A. Neighbor Dead with Upper Layer - Master MGMT Rset shutdown only
B. Neighbor Dead with Lower Layer - Master MGMT Rset shutdown only
C. Delete shutdown active mappings from adjacent processors
[RTR] rClearActiveMap: (Im2)
[RTR] rClearActiveMap: (Im2)
[RTR] rClearActiveMap: (Im2)
[RTR] rClearActiveMap: (Im2)
D. For standbys, delete mappings from active processor and stop updates
E. For critical rset shutdown, delete all their shadows
[RTR] rDelMcastList: (Im2)

Figure 42

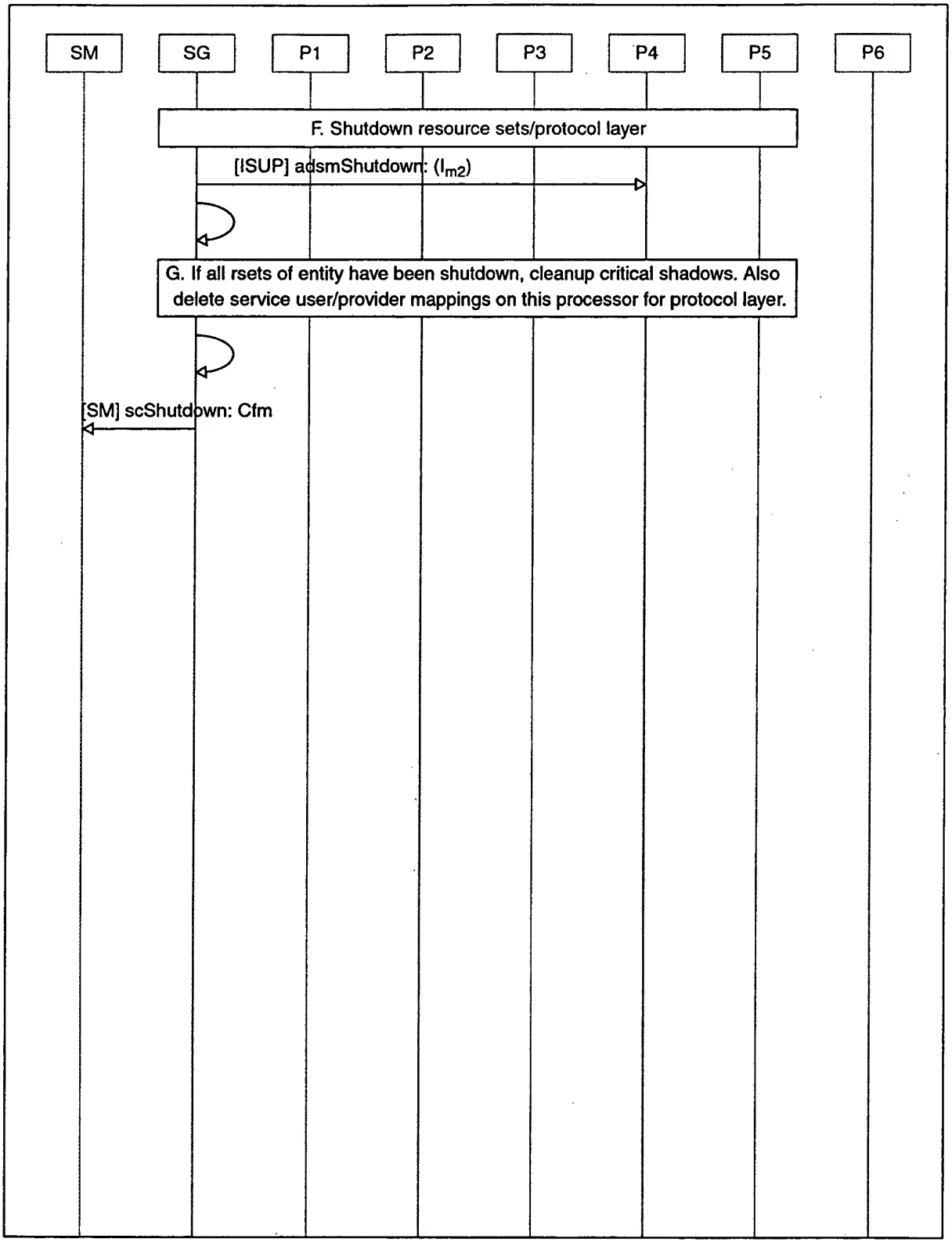
SM
SG
P1
P2
P3
P4
P5
P6
F. Shutdown resource sets/protocol layer
[ISUP] adsmShutdown: (I_{m2})
G. If all rsets of entity have been shutdown, cleanup critical shadows. Also delete service user/provider mappings on this processor for protocol layer.
[SM] scShutdown: Cfm

Figure 43

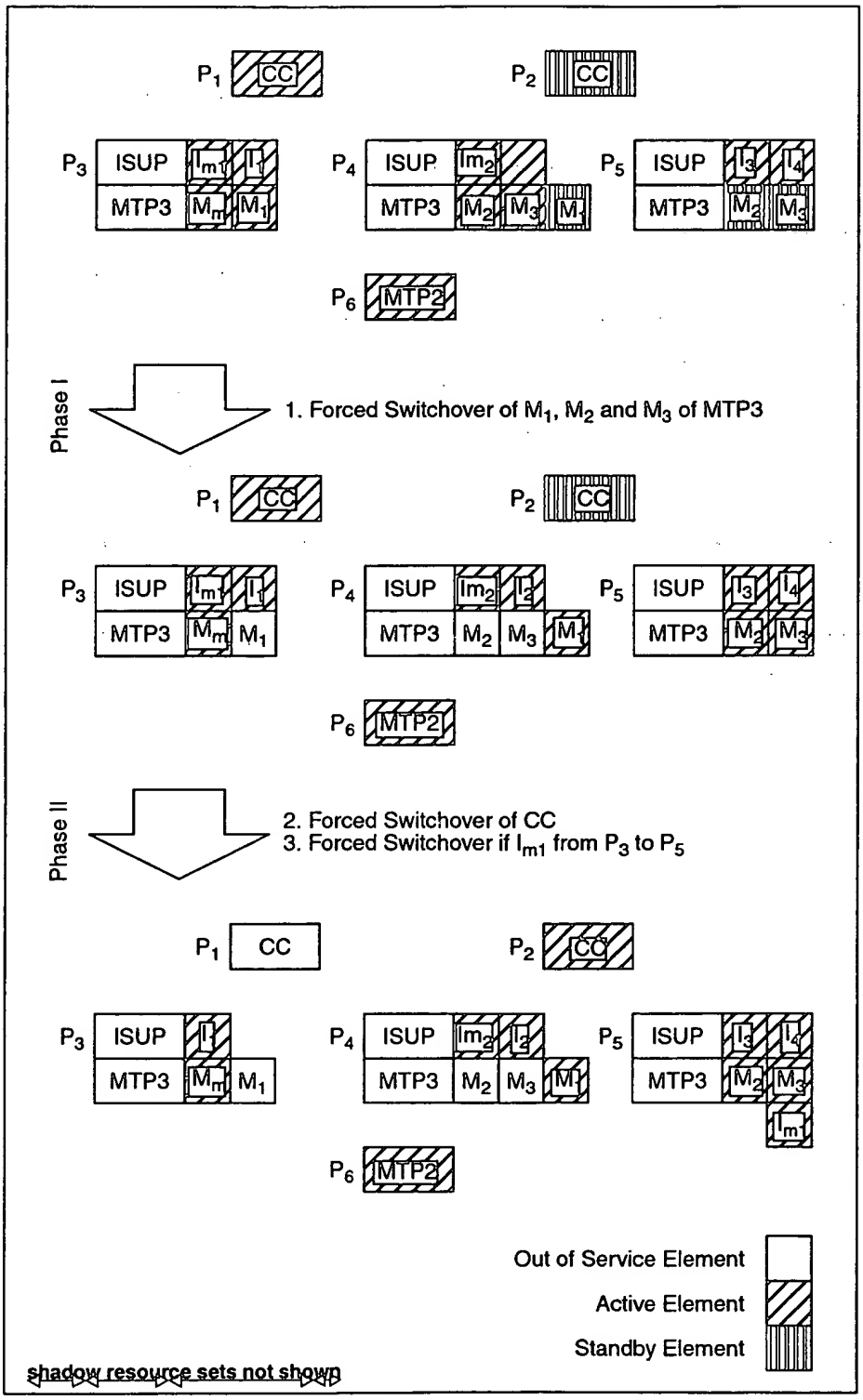
P1
CC
P2
CC
P3
ISUP
MTP3
P4
ISUP
MTP3
P5
ISUP
MTP3
P6
MTP2
Phase I
1. Forced Switchover of M1, M2 and M3 of MTP3
Phase II
2. Forced Switchover of CC
3. Forced Switchover if Im1 from P3 to P5
Out of Service Element
Active Element
Standby Element
shadow resource sets not shown

Figure 44

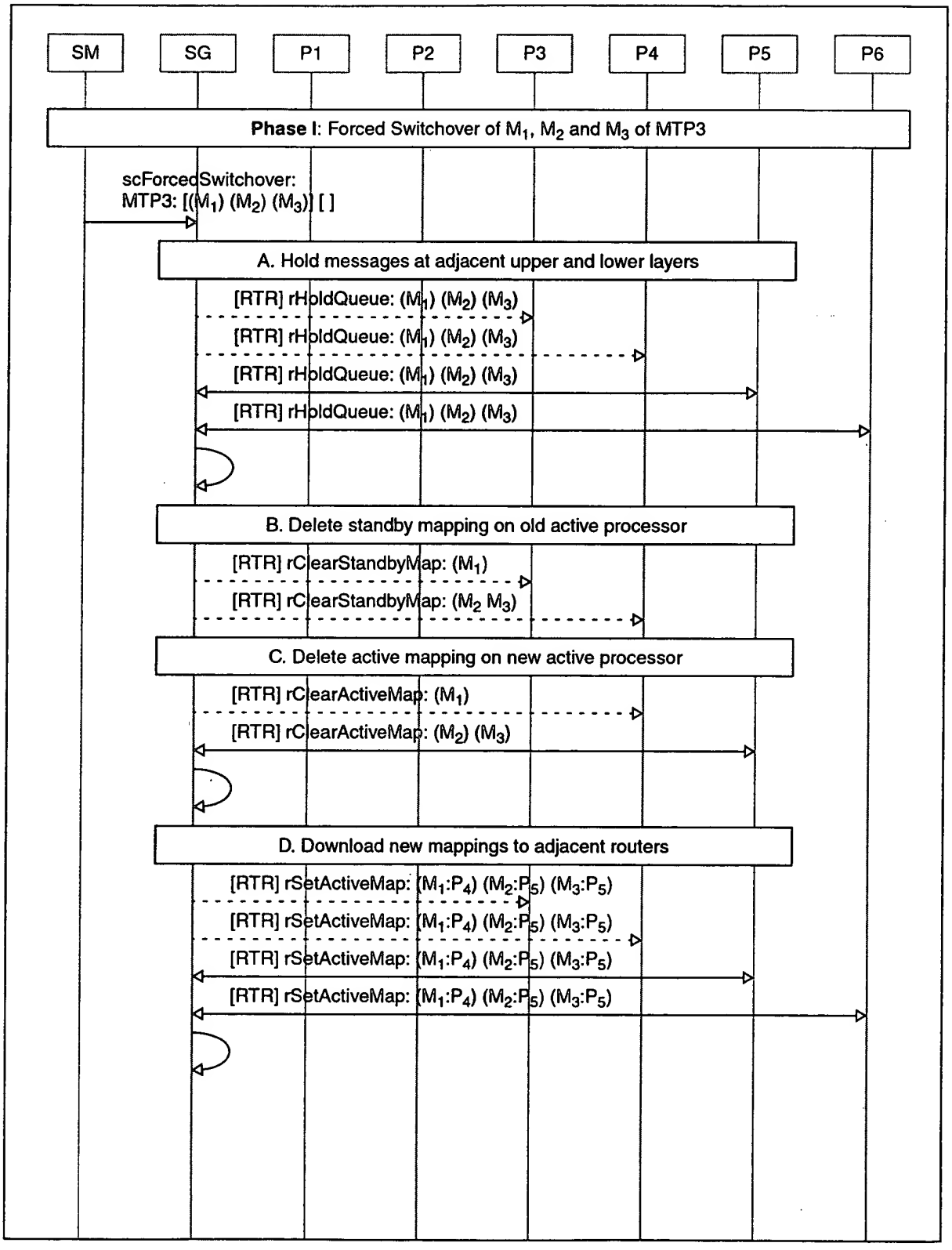
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: Forced Switchover of M1, M2 and M3 of MTP3
scForcedSwitchover:
MTP3: [(M1) (M2) (M3)] [ ]
A. Hold messages at adjacent upper and lower layers
[RTR] rHoldQueue: (M1) (M2) (M3)
[RTR] rHoldQueue: (M1) (M2) (M3)
[RTR] rHoldQueue: (M1) (M2) (M3)
[RTR] rHoldQueue: (M1) (M2) (M3)
B. Delete standby mapping on old active processor
[RTR] rClearStandbyMap: (M1)
[RTR] rClearStandbyMap: (M2 M3)
C. Delete active mapping on new active processor
[RTR] rClearActiveMap: (M1)
[RTR] rClearActiveMap: (M2) (M3)
D. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5)

Figure 45

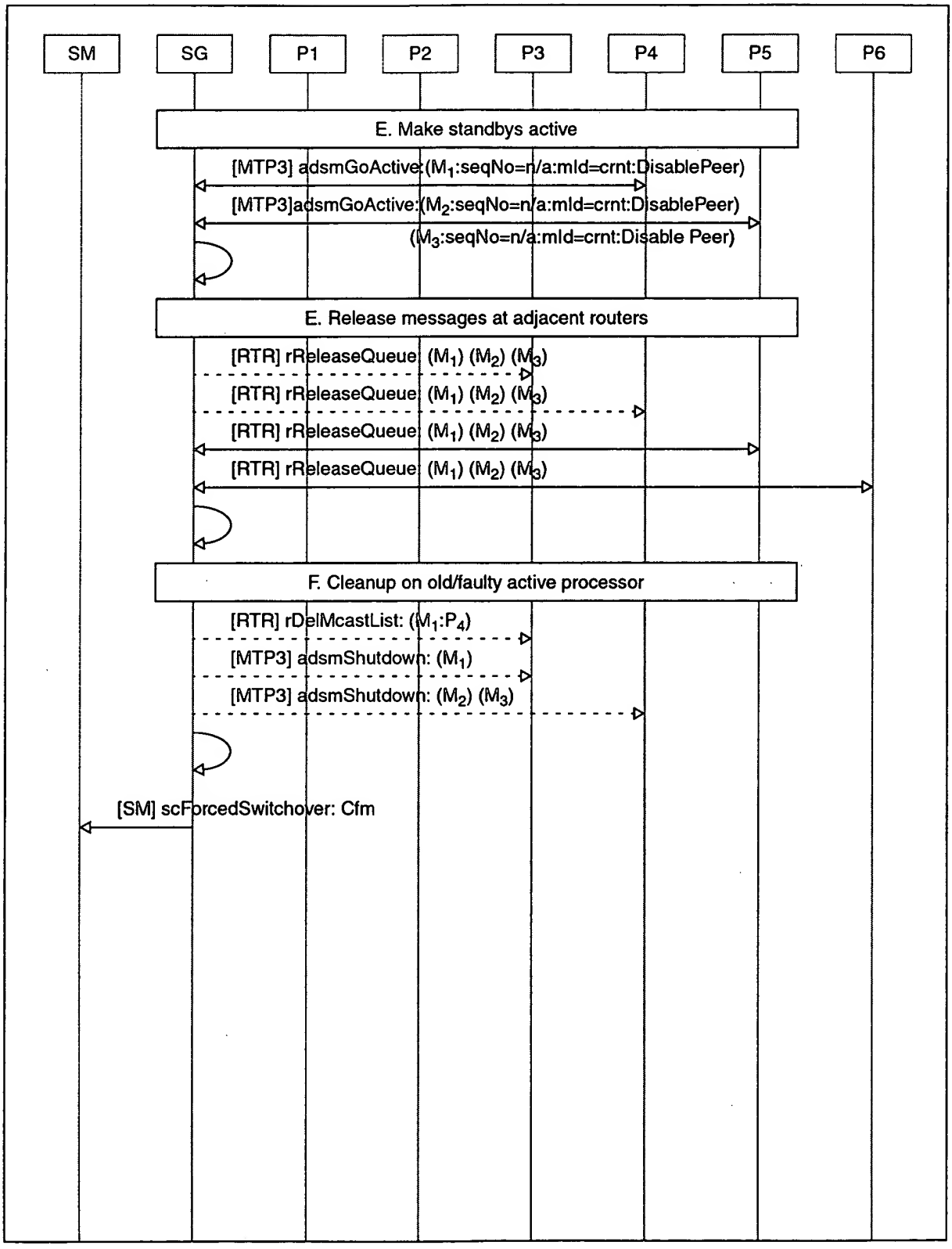
SM
SG
P1
P2
P3
P4
P5
P6
E. Make standbys active
[MTP3] adsmGoActive:(M1:seqNo=n/a:mId=crnt:DisablePeer)
[MTP3]adsmGoActive:(M2:seqNo=n/a:mId=crnt:DisablePeer)
(M3:seqNo=n/a:mId=crnt:Disable Peer)
E. Release messages at adjacent routers
[RTR] rReleaseQueue: (M1) (M2) (M3)
[RTR] rReleaseQueue: (M1) (M2) (M3)
[RTR] rReleaseQueue: (M1) (M2) (M3)
[RTR] rReleaseQueue: (M1) (M2) (M3)
F. Cleanup on old/faulty active processor
[RTR] rDelMcastList: (M1:P4)
[MTP3] adsmShutdown: (M1)
[MTP3] adsmShutdown: (M2) (M3)
[SM] scForcedSwitchover: Cfm

Figure 46

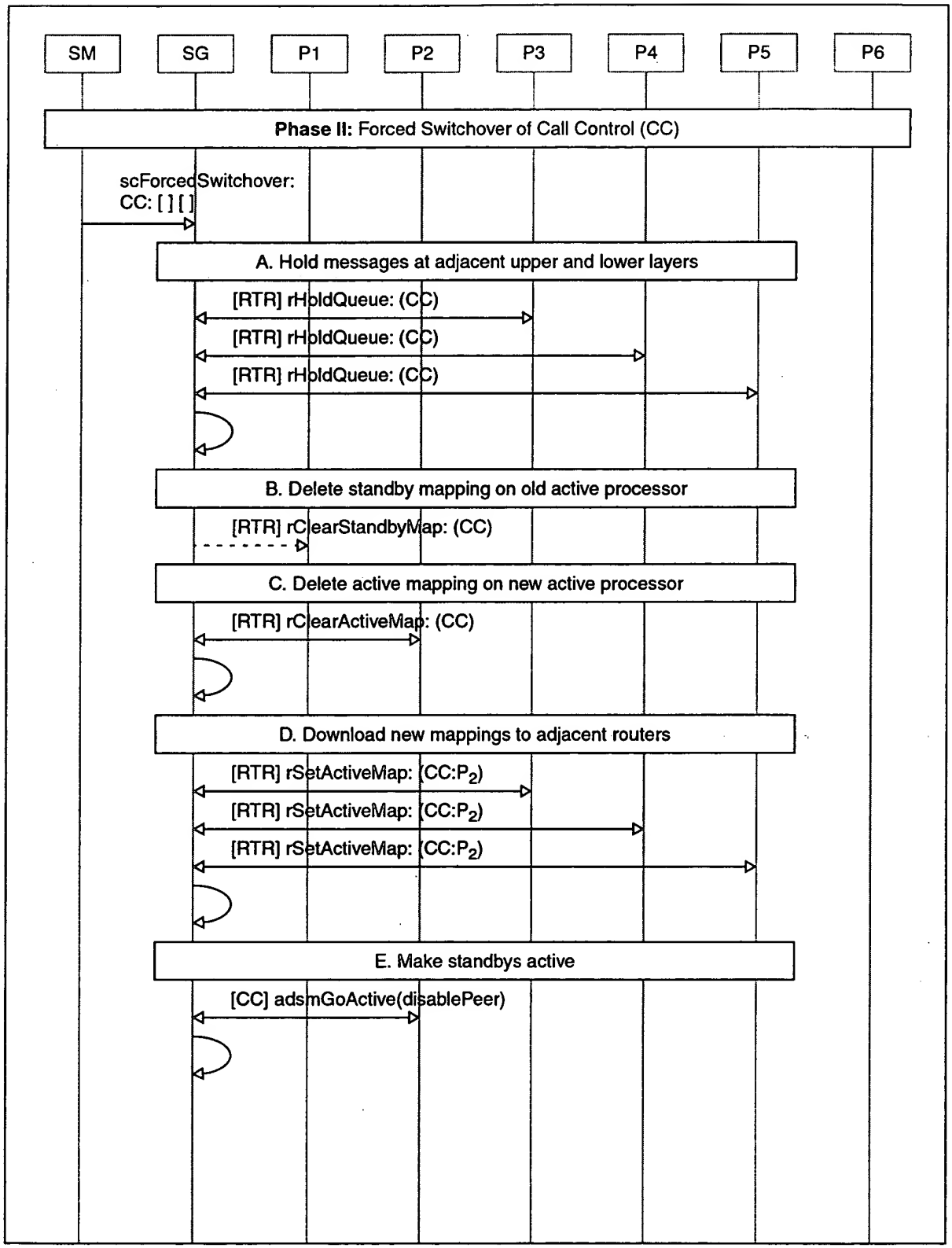
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: Forced Switchover of Call Control (CC)
scForcedSwitchover:
CC: [ ] [ ]
A. Hold messages at adjacent upper and lower layers
[RTR] rHoldQueue: (CC)
[RTR] rHoldQueue: (CC)
[RTR] rHoldQueue: (CC)
B. Delete standby mapping on old active processor
[RTR] rClearStandbyMap: (CC)
C. Delete active mapping on new active processor
[RTR] rClearActiveMap: (CC)
D. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (CC:P2)
[RTR] rSetActiveMap: (CC:P2)
[RTR] rSetActiveMap: (CC:P2)
E. Make standbys active
[CC] adsmGoActive(disablePeer)

Figure 47

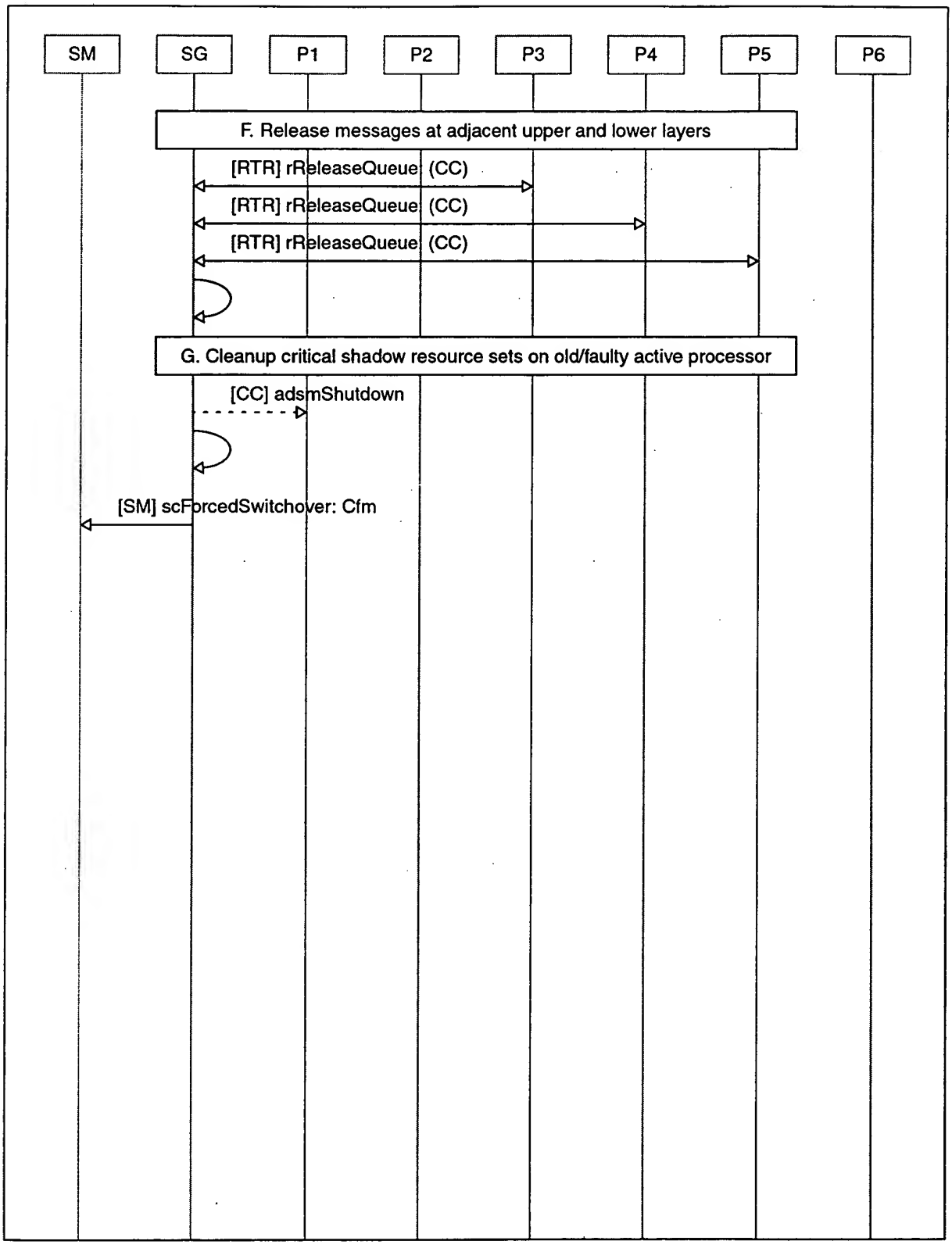
SM
SG
P1
P2
P3
P4
P5
P6
F. Release messages at adjacent upper and lower layers
[RTR] rReleaseQueue (CC)
[RTR] rReleaseQueue (CC)
[RTR] rReleaseQueue (CC)
G. Cleanup critical shadow resource sets on old/faulty active processor
[CC] adsmShutdown
[SM] scForcedSwitchover: Cfm

Figure 48

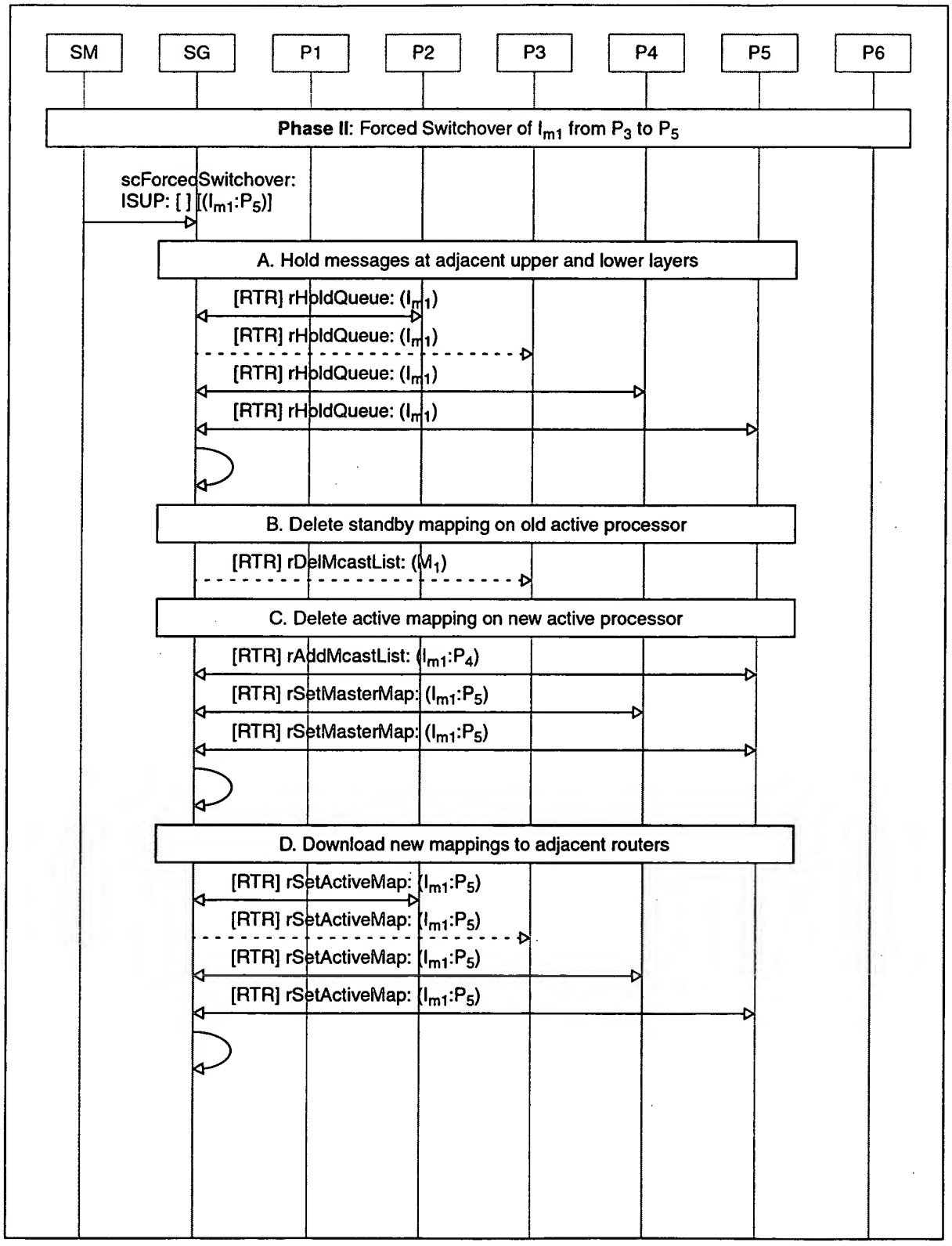
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: Forced Switchover of I_{m1} from P_3 to P_5
scForcedSwitchover:
ISUP: [ ] [(I_{m1}:P_5)]
A. Hold messages at adjacent upper and lower layers
[RTR] rHoldQueue: (I_{m1})
[RTR] rHoldQueue: (I_{m1})
[RTR] rHoldQueue: (I_{m1})
[RTR] rHoldQueue: (I_{m1})
B. Delete standby mapping on old active processor
[RTR] rDelMcastList: (M_1)
C. Delete active mapping on new active processor
[RTR] rAddMcastList: (I_{m1}:P_4)
[RTR] rSetMasterMap: (I_{m1}:P_5)
[RTR] rSetMasterMap: (I_{m1}:P_5)
D. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (I_{m1}:P_5)
[RTR] rSetActiveMap: (I_{m1}:P_5)
[RTR] rSetActiveMap: (I_{m1}:P_5)
[RTR] rSetActiveMap: (I_{m1}:P_5)

Figure 49

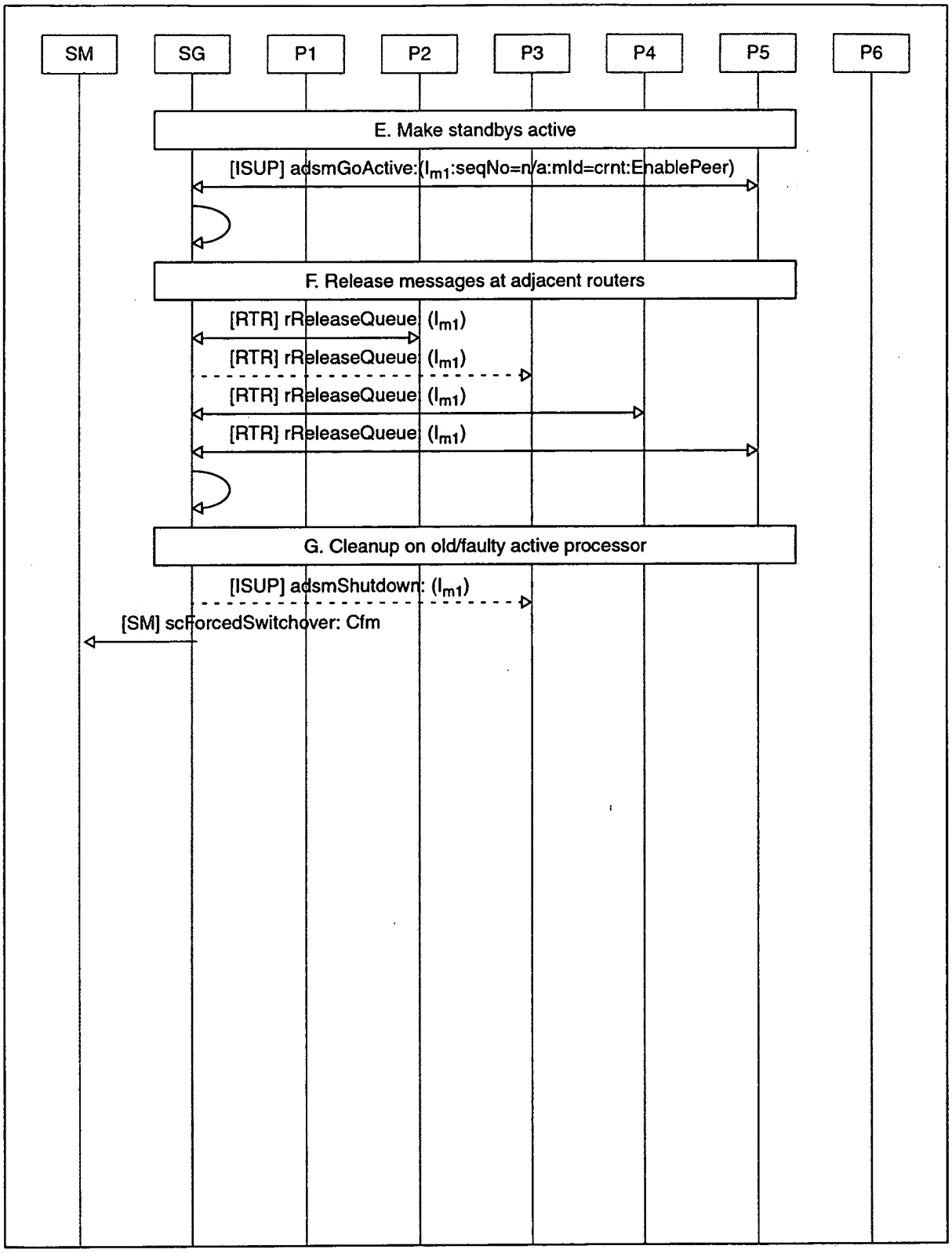
SM
SG
P1
P2
P3
P4
P5
P6
E. Make standbys active
[ISUP] adsmGoActive:(Im1:seqNo=n/a:mId=crnt:EnablePeer)
F. Release messages at adjacent routers
[RTR] rReleaseQueue (Im1)
[RTR] rReleaseQueue (Im1)
[RTR] rReleaseQueue (Im1)
[RTR] rReleaseQueue (Im1)
G. Cleanup on old/faulty active processor
[ISUP] adsmShutdown: (Im1)
[SM] scForcedSwitchover: Cfm

Figure 50

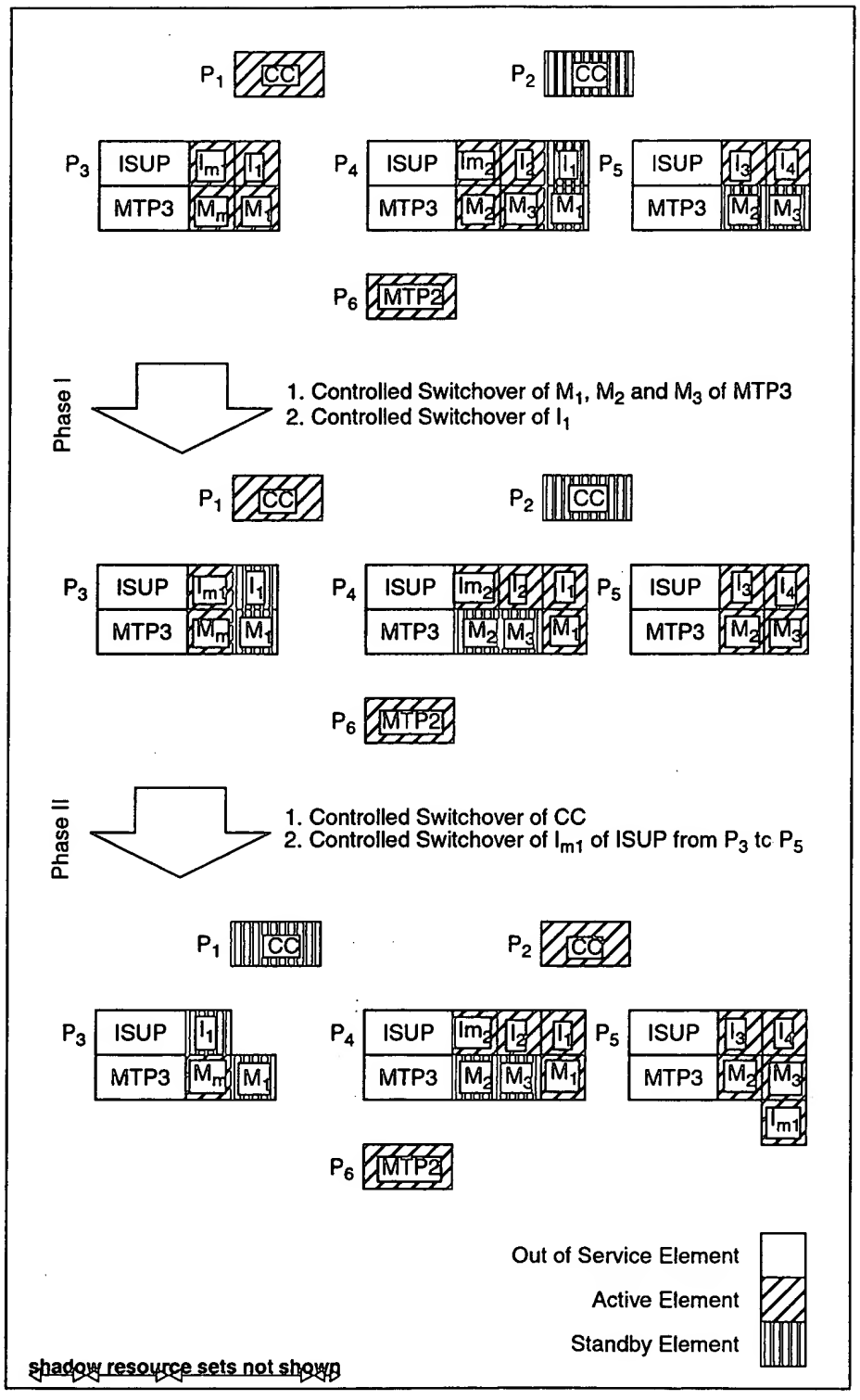
P1
CC
P2
CC
P3
ISUP
MTP3
P4
ISUP
MTP3
P5
ISUP
MTP3
P6
MTP2
Phase I
1. Controlled Switchover of M1, M2 and M3 of MTP3
2. Controlled Switchover of I1
Phase II
1. Controlled Switchover of CC
2. Controlled Switchover of Im1 of ISUP from P3 tc P5
Out of Service Element
Active Element
Standby Element
shadow resource sets not shown

Figure 51

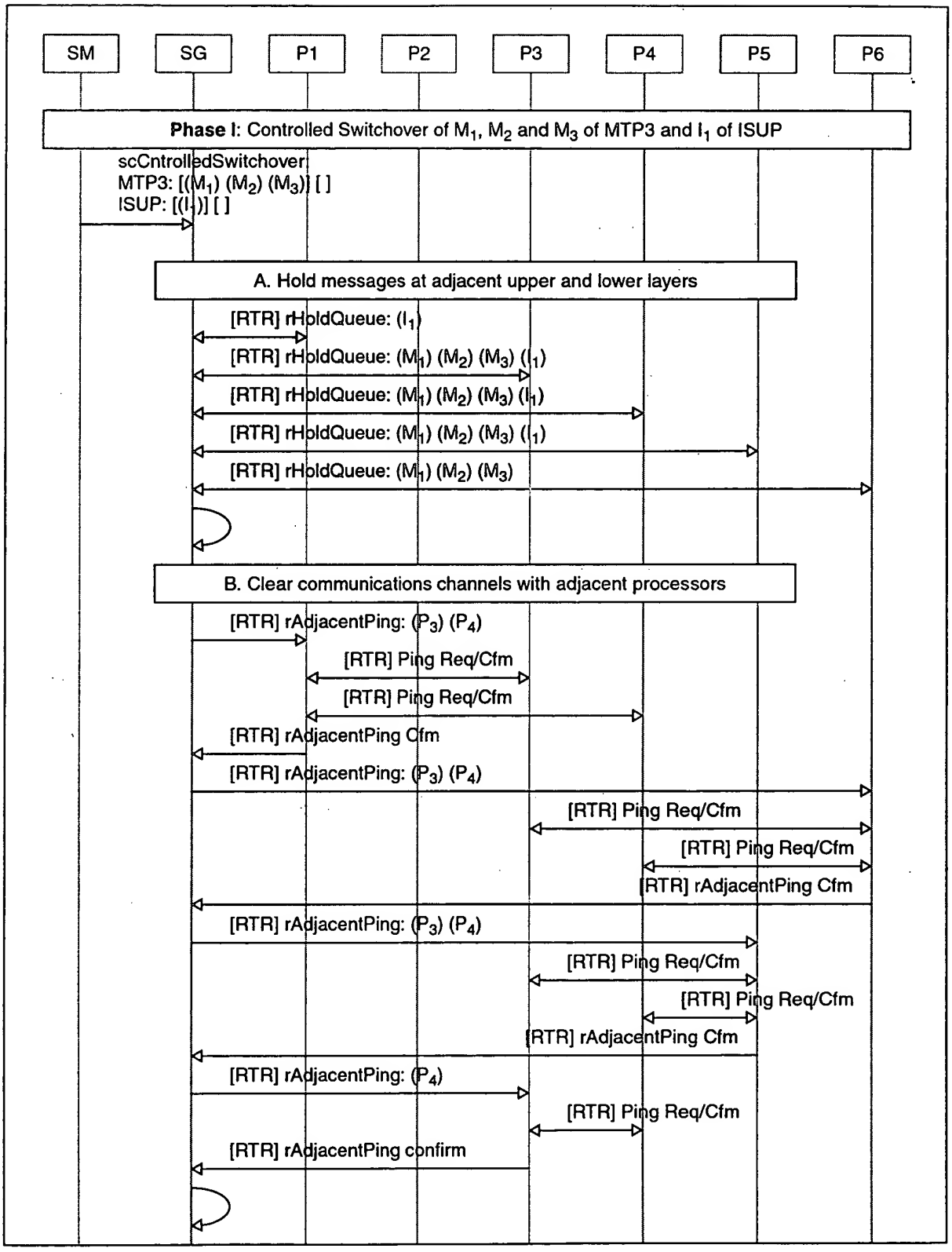
SM
SG
P1
P2
P3
P4
P5
P6
Phase I: Controlled Switchover of M1, M2 and M3 of MTP3 and I1 of ISUP
scCntrolledSwitchover
MTP3: [(M1) (M2) (M3)] [ ]
ISUP: [(I1)] [ ]
A. Hold messages at adjacent upper and lower layers
[RTR] rHoldQueue: (I1)
[RTR] rHoldQueue: (M1) (M2) (M3) (I1)
[RTR] rHoldQueue: (M1) (M2) (M3) (I1)
[RTR] rHoldQueue: (M1) (M2) (M3) (I1)
[RTR] rHoldQueue: (M1) (M2) (M3)
B. Clear communications channels with adjacent processors
[RTR] rAdjacentPing: (P3) (P4)
[RTR] Ping Req/Cfm
[RTR] Ping Req/Cfm
[RTR] rAdjacentPing Cfm
[RTR] rAdjacentPing: (P3) (P4)
[RTR] Ping Req/Cfm
[RTR] Ping Req/Cfm
[RTR] rAdjacentPing Cfm
[RTR] rAdjacentPing: (P3) (P4)
[RTR] Ping Req/Cfm
[RTR] Ping Req/Cfm
[RTR] rAdjacentPing Cfm
[RTR] rAdjacentPing: (P4)
[RTR] Ping Req/Cfm
[RTR] rAdjacentPing confirm

Figure 52

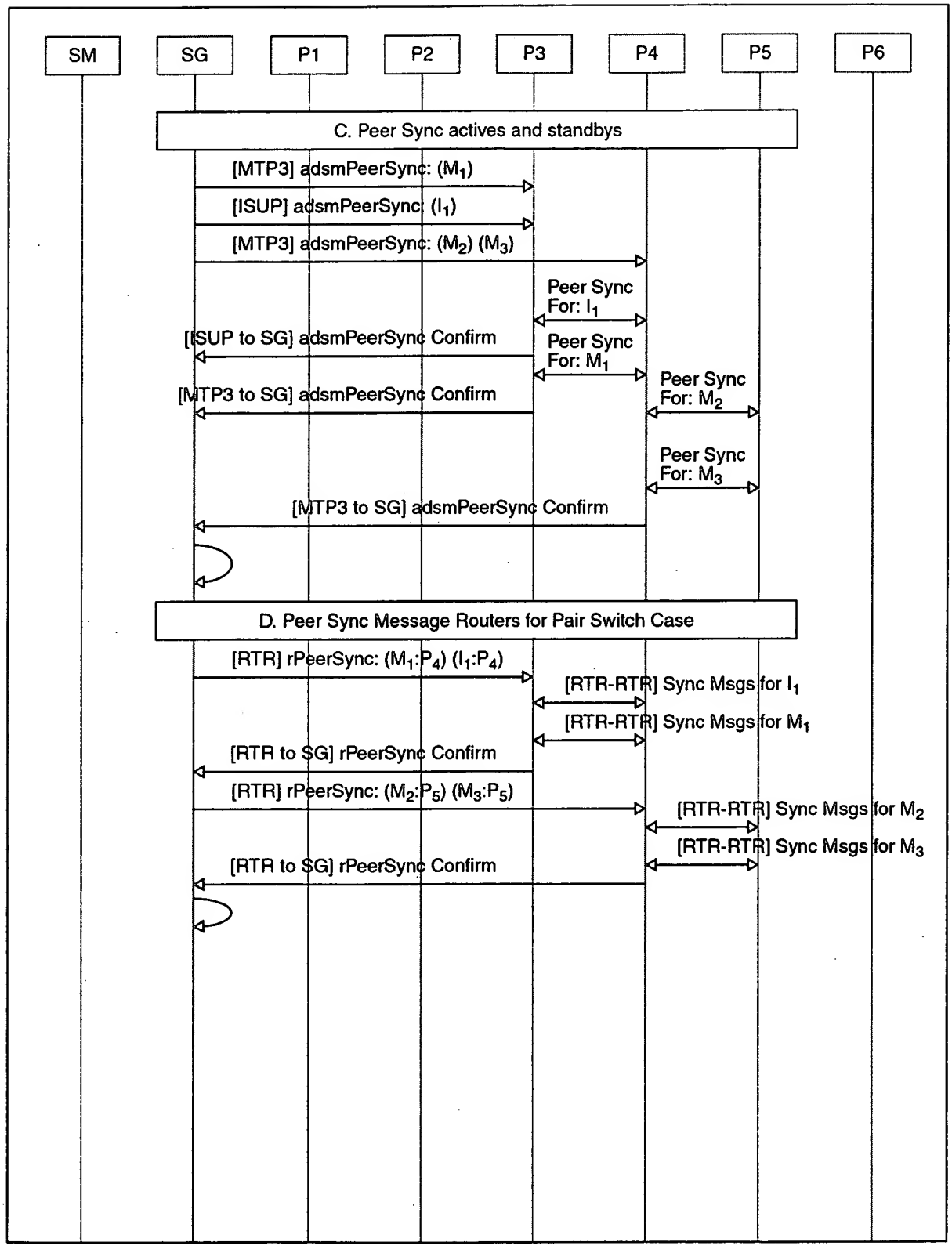
SM
SG
P1
P2
P3
P4
P5
P6
C. Peer Sync actives and standbys
[MTP3] adsmPeerSync: (M1)
[ISUP] adsmPeerSync: (I1)
[MTP3] adsmPeerSync: (M2) (M3)
Peer Sync For: I1
[ISUP to SG] adsmPeerSync Confirm
Peer Sync For: M1
Peer Sync For: M2
[MTP3 to SG] adsmPeerSync Confirm
Peer Sync For: M3
[MTP3 to SG] adsmPeerSync Confirm
D. Peer Sync Message Routers for Pair Switch Case
[RTR] rPeerSync: (M1:P4) (I1:P4)
[RTR-RTR] Sync Msgs for I1
[RTR-RTR] Sync Msgs for M1
[RTR to SG] rPeerSync Confirm
[RTR] rPeerSync: (M2:P5) (M3:P5)
[RTR-RTR] Sync Msgs for M2
[RTR-RTR] Sync Msgs for M3
[RTR to SG] rPeerSync Confirm

Figure 53

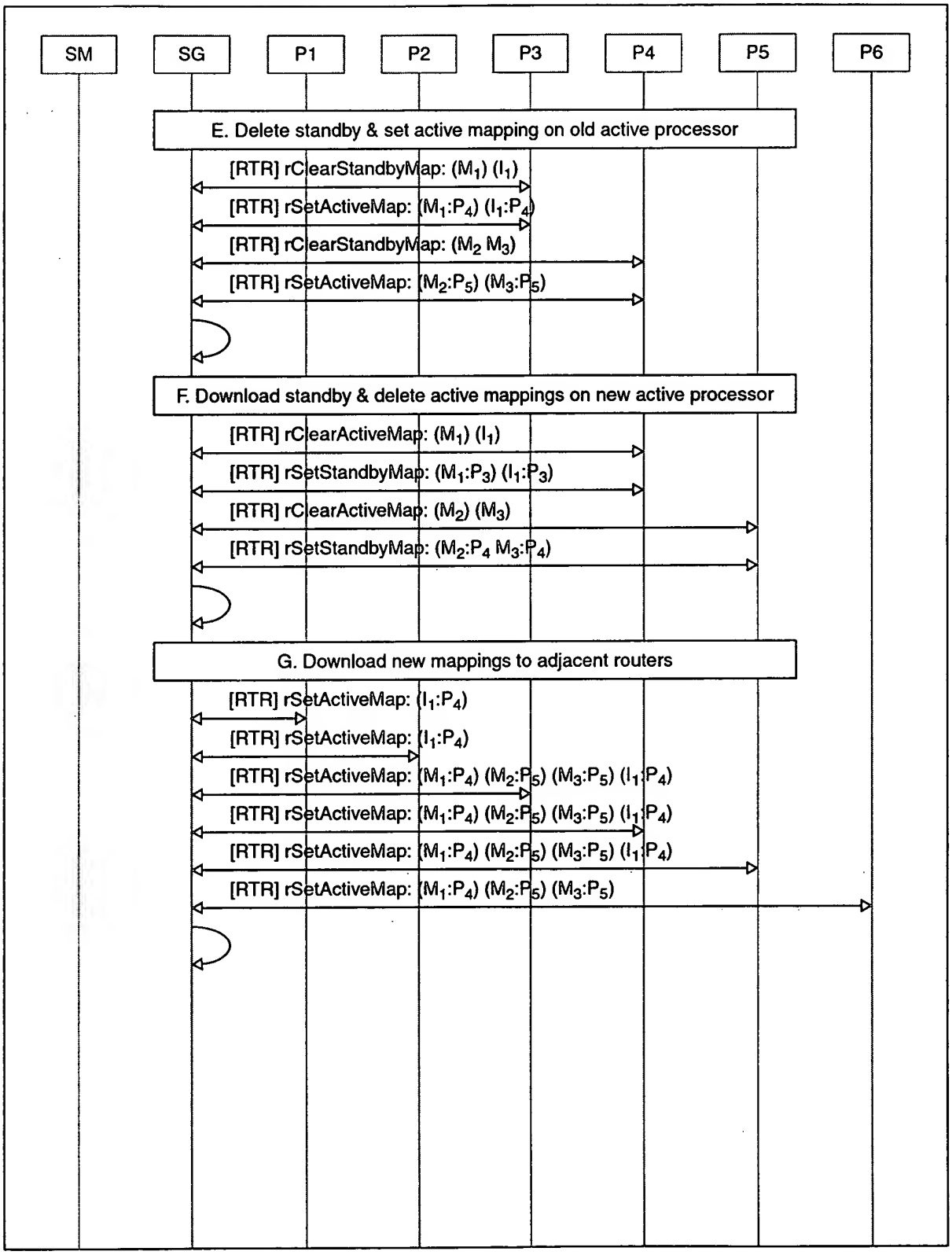
SM
SG
P1
P2
P3
P4
P5
P6
E. Delete standby & set active mapping on old active processor
[RTR] rClearStandbyMap: (M1) (I1)
[RTR] rSetActiveMap: (M1:P4) (I1:P4)
[RTR] rClearStandbyMap: (M2 M3)
[RTR] rSetActiveMap: (M2:P5) (M3:P5)
F. Download standby & delete active mappings on new active processor
[RTR] rClearActiveMap: (M1) (I1)
[RTR] rSetStandbyMap: (M1:P3) (I1:P3)
[RTR] rClearActiveMap: (M2) (M3)
[RTR] rSetStandbyMap: (M2:P4 M3:P4)
G. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (I1:P4)
[RTR] rSetActiveMap: (I1:P4)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5) (I1:P4)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5) (I1:P4)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5) (I1:P4)
[RTR] rSetActiveMap: (M1:P4) (M2:P5) (M3:P5)

Figure 54

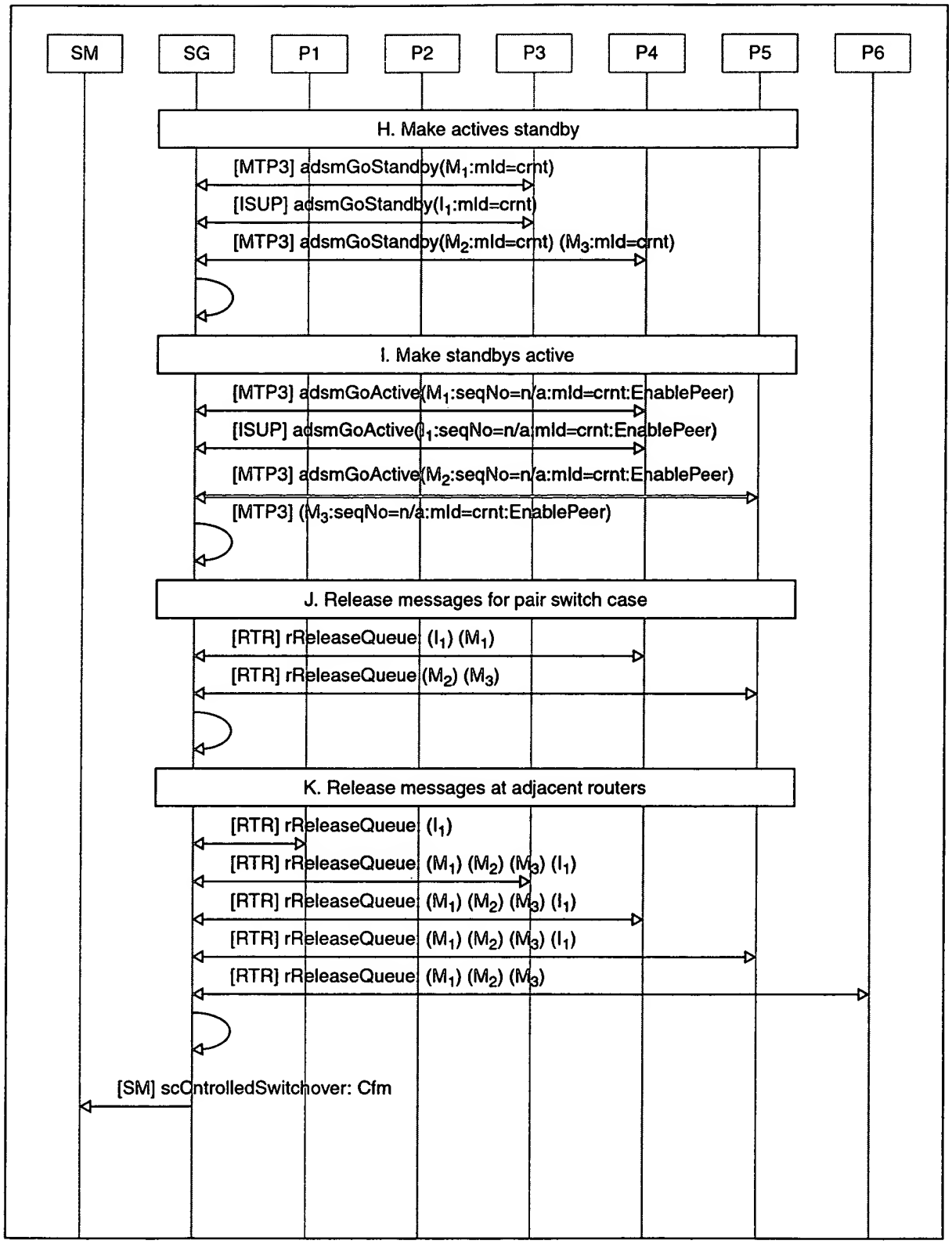
SM
SG
P1
P2
P3
P4
P5
P6
H. Make actives standby
[MTP3] adsmGoStandby(M1:mId=crnt)
[ISUP] adsmGoStandby(I1:mId=crnt)
[MTP3] adsmGoStandby(M2:mId=crnt) (M3:mId=crnt)
I. Make standbys active
[MTP3] adsmGoActive(M1:seqNo=n/a:mId=crnt:EnablePeer)
[ISUP] adsmGoActive(I1:seqNo=n/a:mId=crnt:EnablePeer)
[MTP3] adsmGoActive(M2:seqNo=n/a:mId=crnt:EnablePeer)
[MTP3] (M3:seqNo=n/a:mId=crnt:EnablePeer)
J. Release messages for pair switch case
[RTR] rReleaseQueue (I1) (M1)
[RTR] rReleaseQueue (M2) (M3)
K. Release messages at adjacent routers
[RTR] rReleaseQueue (I1)
[RTR] rReleaseQueue (M1) (M2) (M3) (I1)
[RTR] rReleaseQueue (M1) (M2) (M3) (I1)
[RTR] rReleaseQueue (M1) (M2) (M3) (I1)
[RTR] rReleaseQueue (M1) (M2) (M3)
[SM] scControlledSwitchover: Cfm

Figure 55

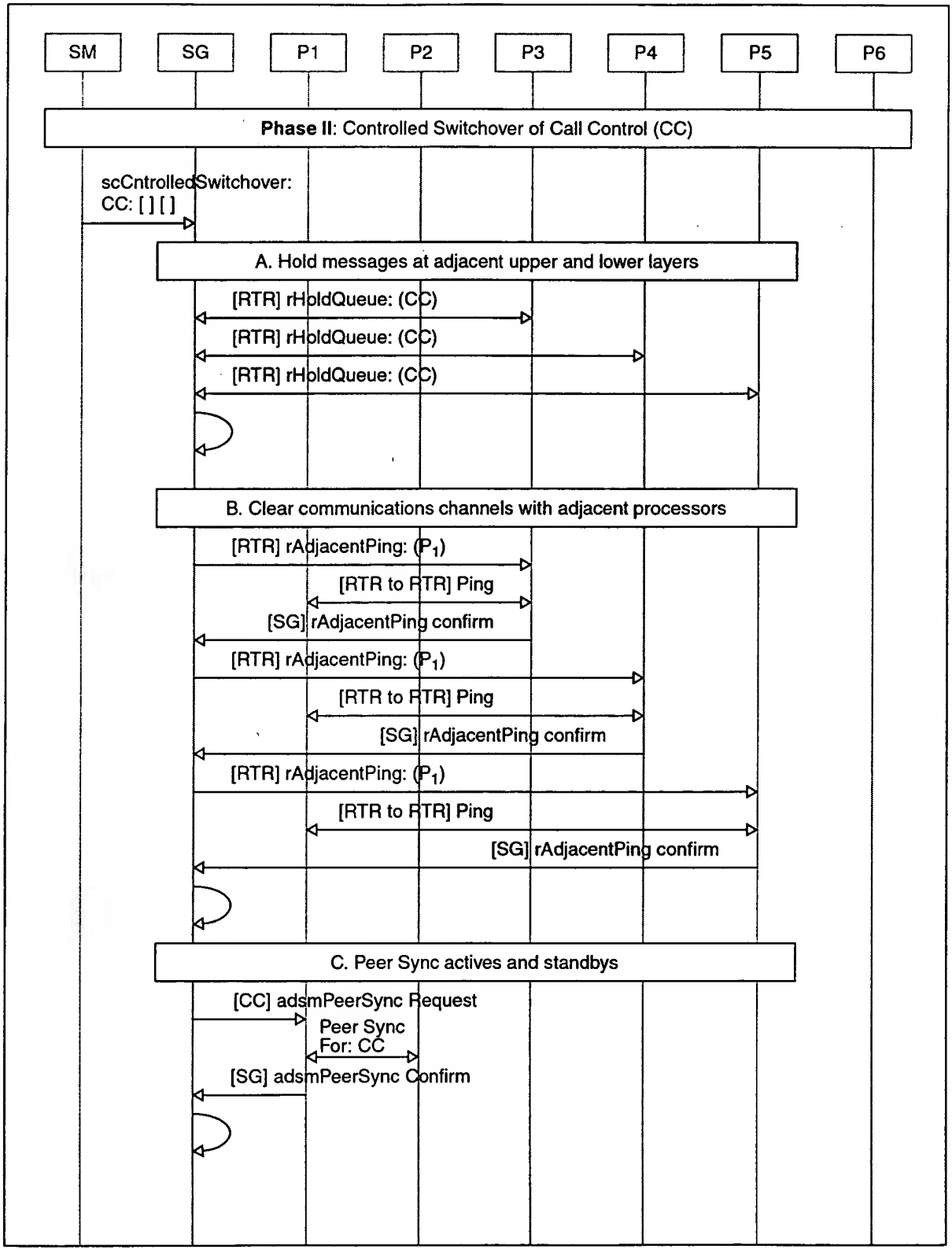
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: Controlled Switchover of Call Control (CC)
scCntrolledSwitchover:
CC: [ ] [ ]
A. Hold messages at adjacent upper and lower layers
[RTR] rHoldQueue: (CC)
[RTR] rHoldQueue: (CC)
[RTR] rHoldQueue: (CC)
B. Clear communications channels with adjacent processors
[RTR] rAdjacentPing: (P1)
[RTR to RTR] Ping
[SG] rAdjacentPing confirm
[RTR] rAdjacentPing: (P1)
[RTR to RTR] Ping
[SG] rAdjacentPing confirm
[RTR] rAdjacentPing: (P1)
[RTR to RTR] Ping
[SG] rAdjacentPing confirm
C. Peer Sync actives and standbys
[CC] adsmPeerSync Request
Peer Sync
For: CC
[SG] adsmPeerSync Confirm

Figure 56

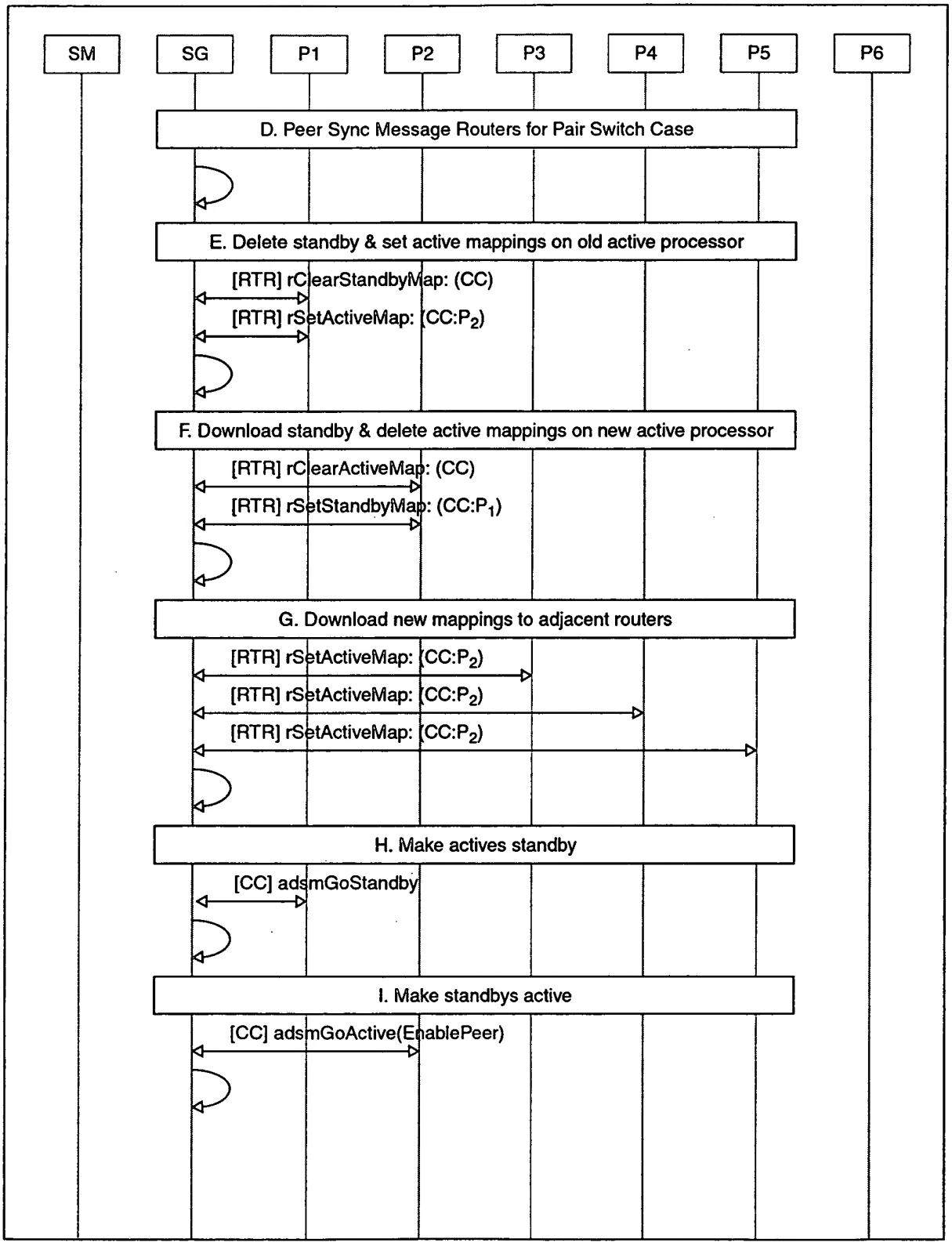
SM
SG
P1
P2
P3
P4
P5
P6
D. Peer Sync Message Routers for Pair Switch Case
E. Delete standby & set active mappings on old active processor
[RTR] rClearStandbyMap: (CC)
[RTR] rSetActiveMap: (CC:P2)
F. Download standby & delete active mappings on new active processor
[RTR] rClearActiveMap: (CC)
[RTR] rSetStandbyMap: (CC:P1)
G. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (CC:P2)
[RTR] rSetActiveMap: (CC:P2)
[RTR] rSetActiveMap: (CC:P2)
H. Make actives standby
[CC] adsmGoStandby
I. Make standbys active
[CC] adsmGoActive(EnablePeer)

Figure 57

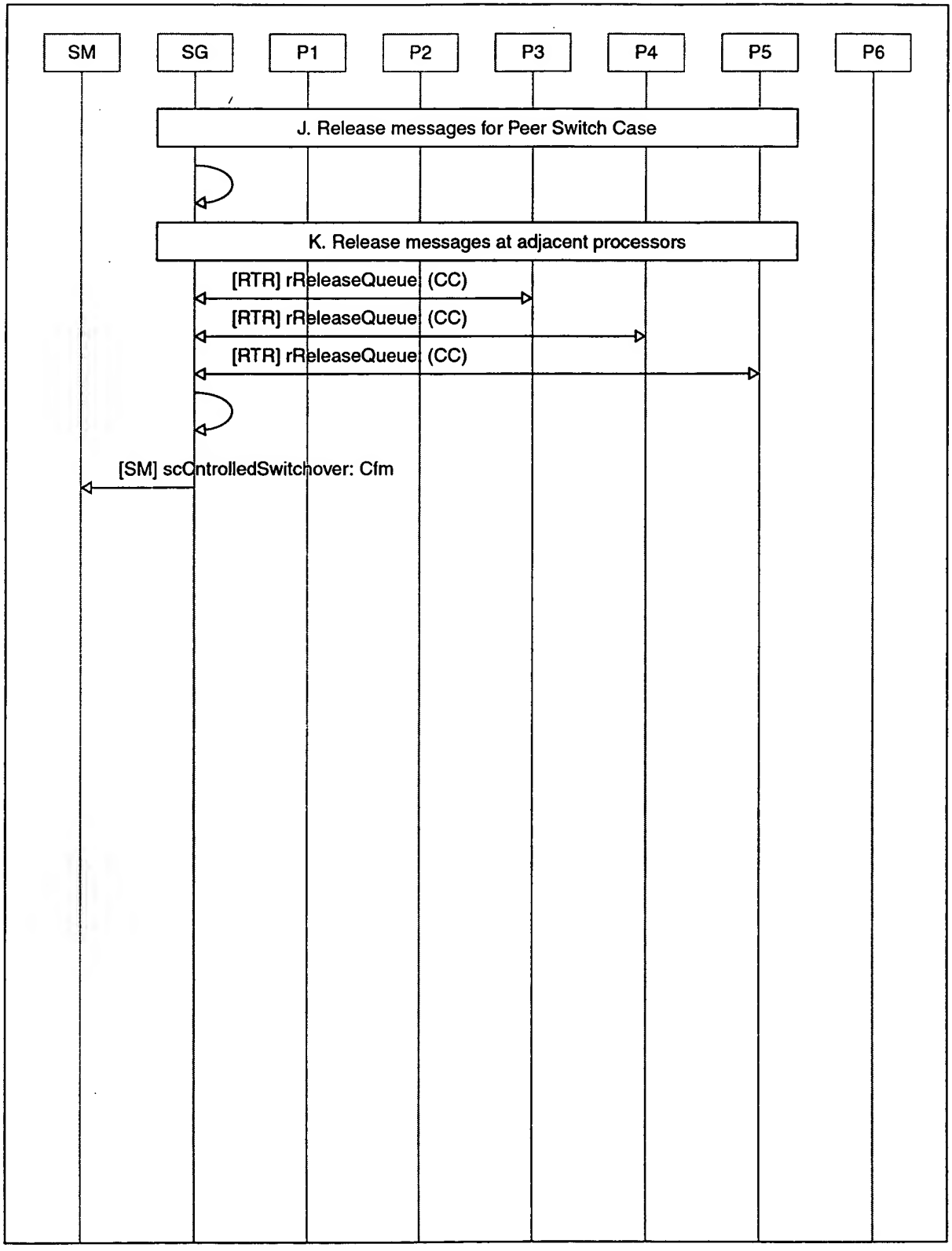
SM
SG
P1
P2
P3
P4
P5
P6
J. Release messages for Peer Switch Case
K. Release messages at adjacent processors
[RTR] rReleaseQueue (CC)
[RTR] rReleaseQueue (CC)
[RTR] rReleaseQueue (CC)
[SM] scControlledSwitchover: Cfm

Figure 58

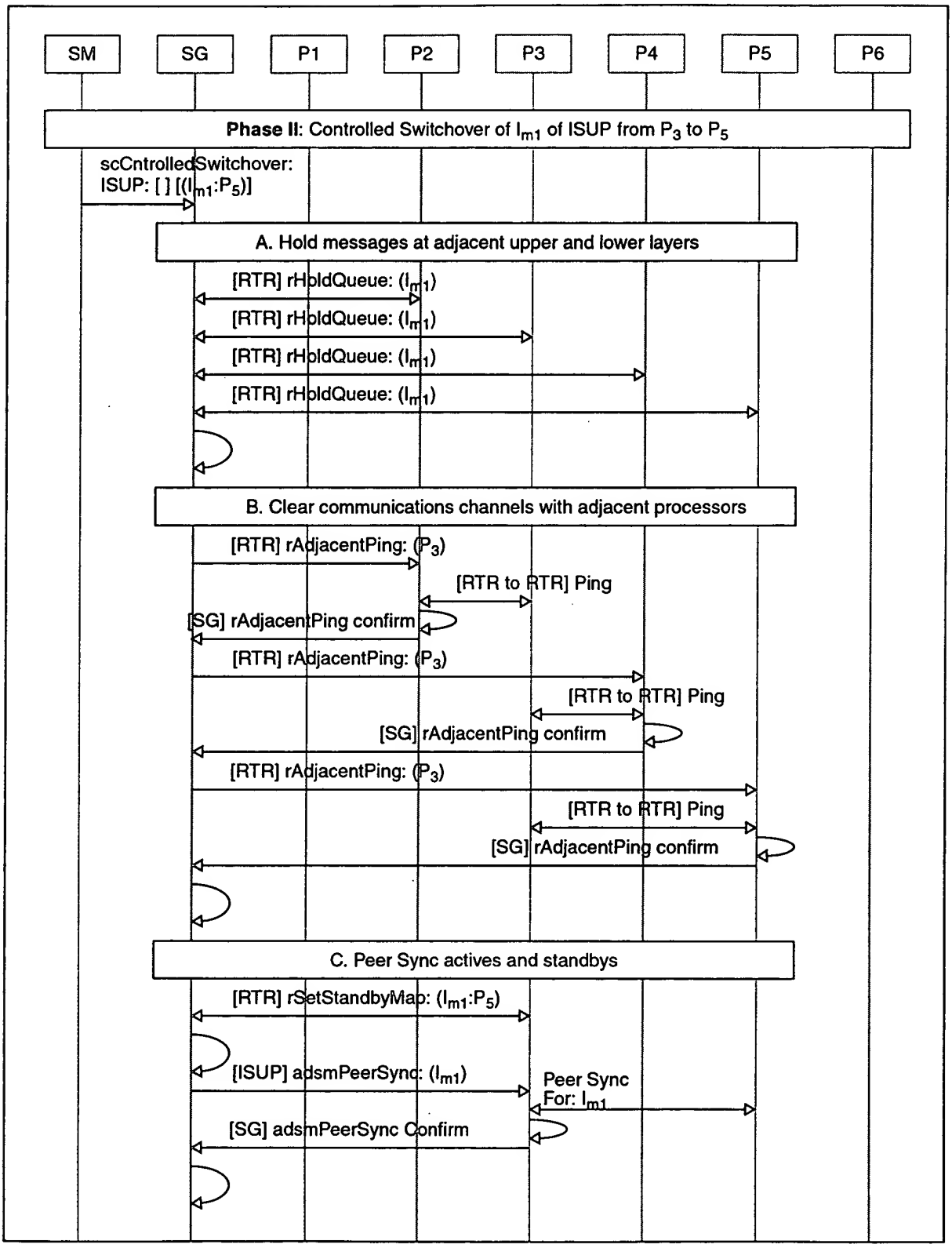
SM
SG
P1
P2
P3
P4
P5
P6
Phase II: Controlled Switchover of I_{m1} of ISUP from P_3 to P_5
scCntrolledSwitchover:
ISUP: [ ] [(I_{m1}:P_5)]
A. Hold messages at adjacent upper and lower layers
[RTR] rHoldQueue: (I_{m1})
[RTR] rHoldQueue: (I_{m1})
[RTR] rHoldQueue: (I_{m1})
[RTR] rHoldQueue: (I_{m1})
B. Clear communications channels with adjacent processors
[RTR] rAdjacentPing: (P_3)
[RTR to RTR] Ping
[SG] rAdjacentPing confirm
[RTR] rAdjacentPing: (P_3)
[RTR to RTR] Ping
[SG] rAdjacentPing confirm
[RTR] rAdjacentPing: (P_3)
[RTR to RTR] Ping
[SG] rAdjacentPing confirm
C. Peer Sync actives and standbys
[RTR] rSetStandbyMap: (I_{m1}:P_5)
[ISUP] adsmPeerSync: (I_{m1})
Peer Sync
For: I_{m1}
[SG] adsmPeerSync Confirm

Figure 59

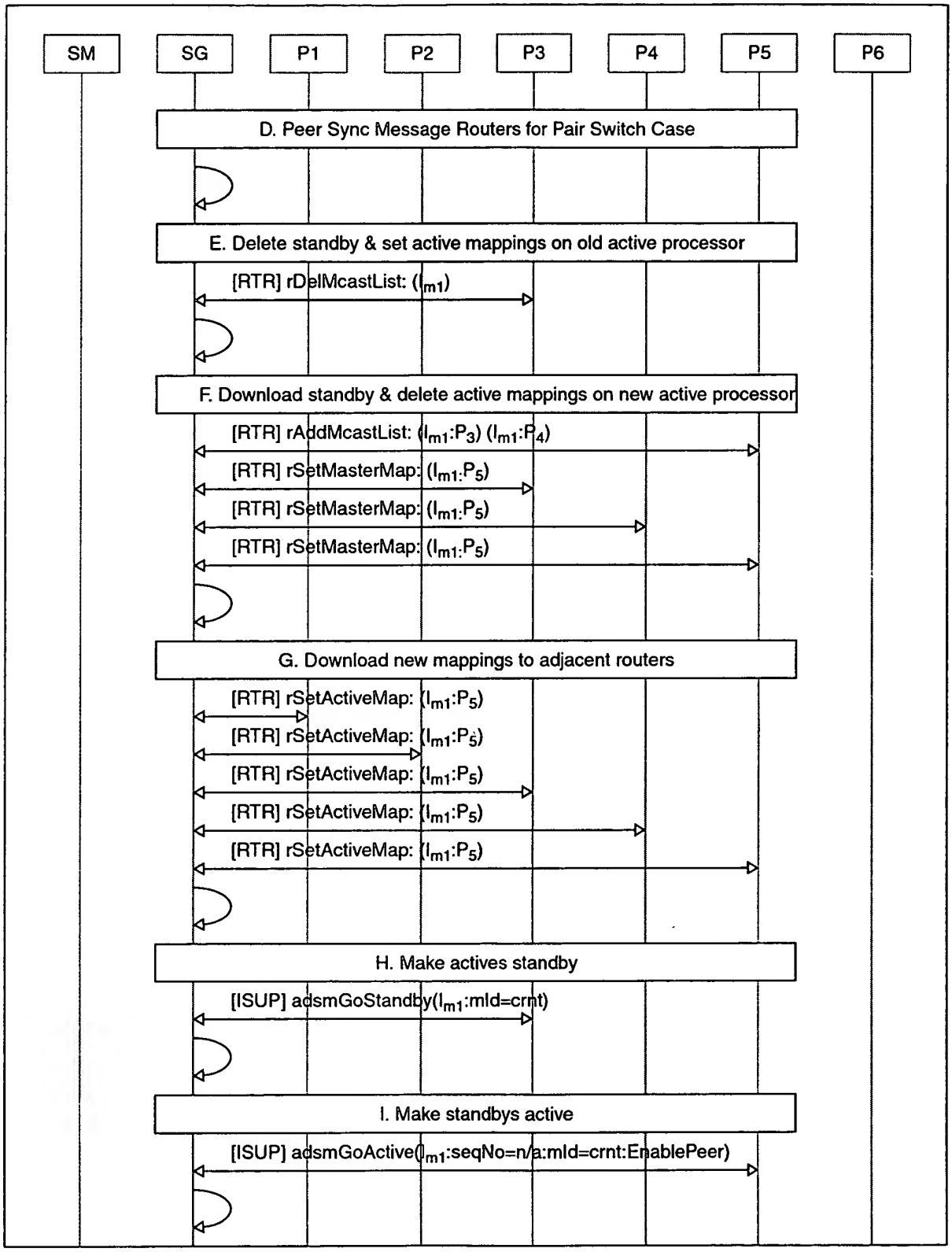
SM
SG
P1
P2
P3
P4
P5
P6
D. Peer Sync Message Routers for Pair Switch Case
E. Delete standby & set active mappings on old active processor
[RTR] rDelMcastList: (l_{m1})
F. Download standby & delete active mappings on new active processor
[RTR] rAddMcastList: (l_{m1}:P_3) (l_{m1}:P_4)
[RTR] rSetMasterMap: (l_{m1}:P_5)
[RTR] rSetMasterMap: (l_{m1}:P_5)
[RTR] rSetMasterMap: (l_{m1}:P_5)
G. Download new mappings to adjacent routers
[RTR] rSetActiveMap: (l_{m1}:P_5)
[RTR] rSetActiveMap: (l_{m1}:P_5)
[RTR] rSetActiveMap: (l_{m1}:P_5)
[RTR] rSetActiveMap: (l_{m1}:P_5)
[RTR] rSetActiveMap: (l_{m1}:P_5)
H. Make actives standby
[ISUP] adsmGoStandby(l_{m1}:mId=crnt)
I. Make standbys active
[ISUP] adsmGoActive(l_{m1}:seqNo=n/a:mId=crnt:EnablePeer)

Figure 60

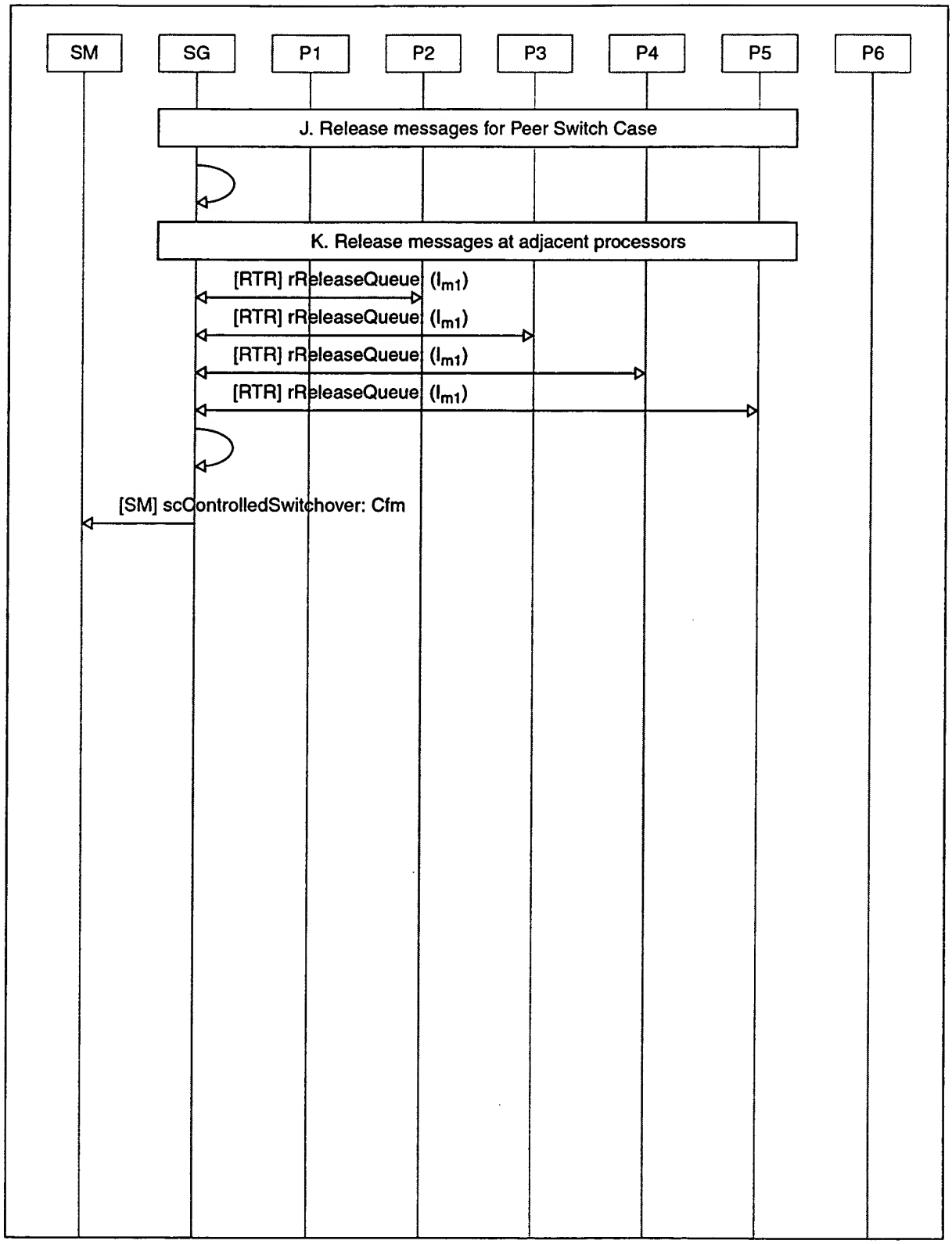
SM
SG
P1
P2
P3
P4
P5
P6
J. Release messages for Peer Switch Case
K. Release messages at adjacent processors
[RTR] rReleaseQueue (lm1)
[RTR] rReleaseQueue (lm1)
[RTR] rReleaseQueue (lm1)
[RTR] rReleaseQueue (lm1)
[SM] scControlledSwitchover: Cfm

Figure 61

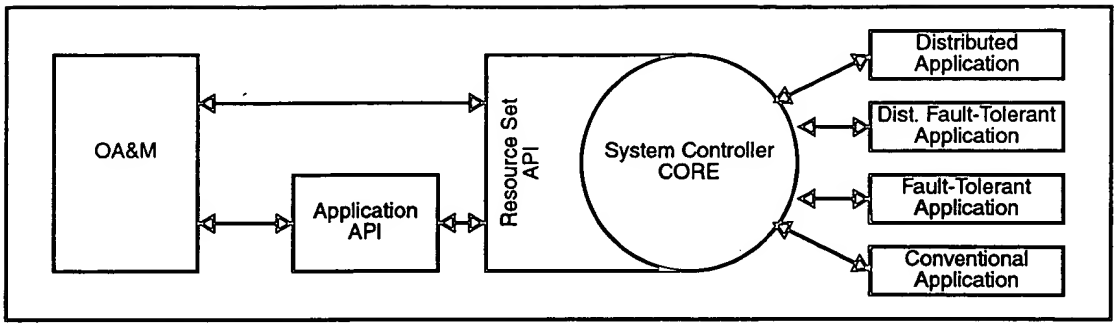

Figure 62

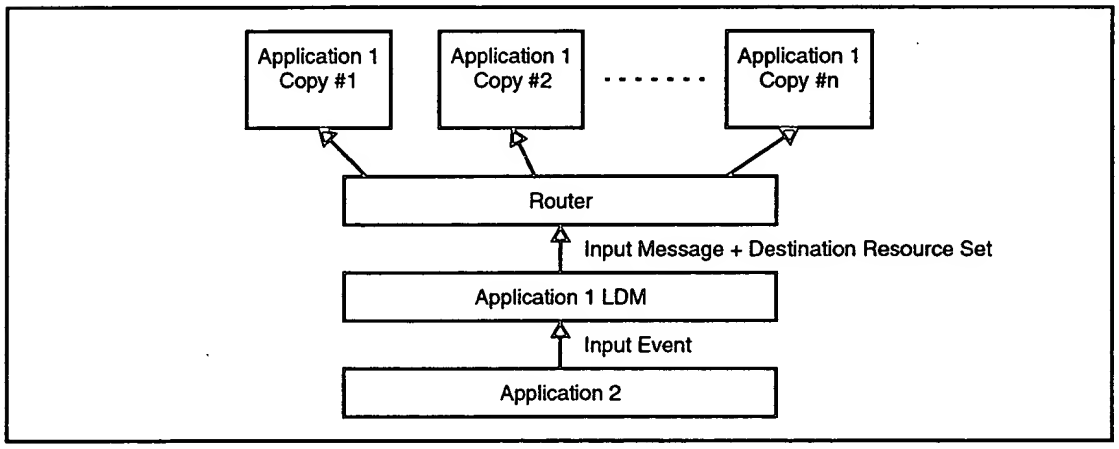

Figure 63

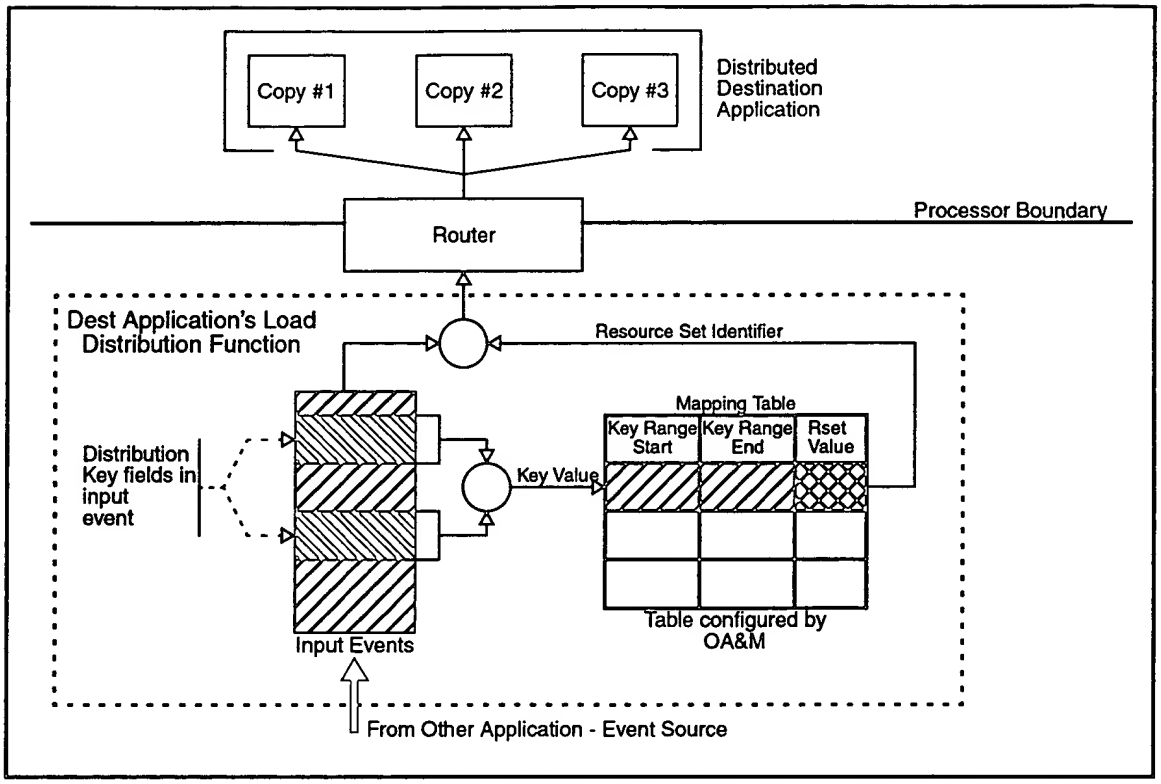

Figure 64

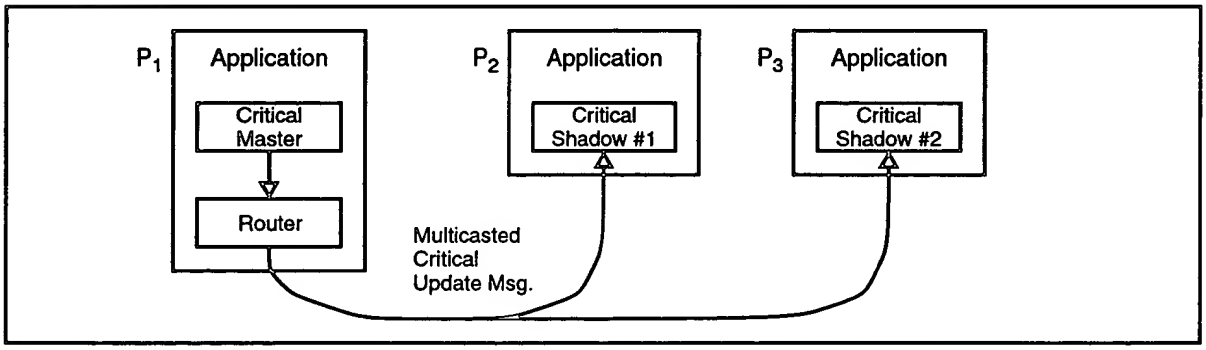

Figure 65

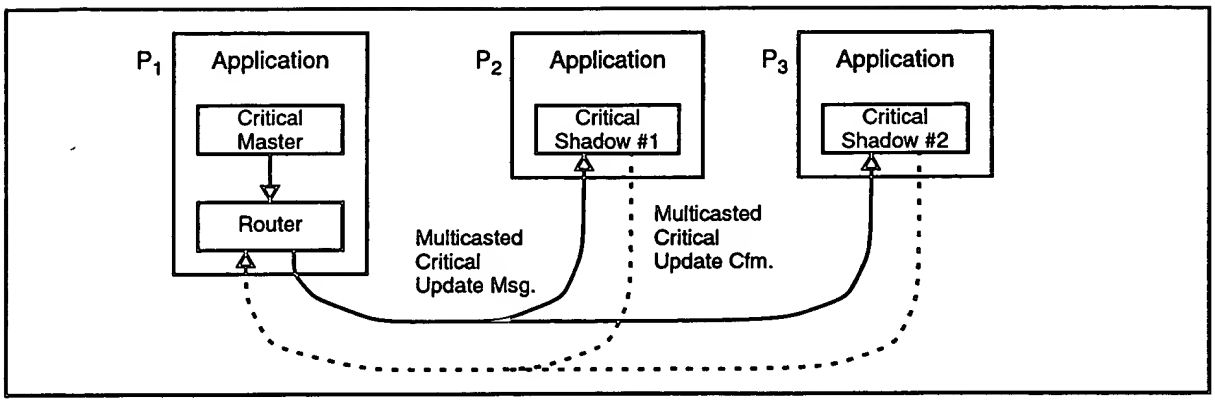
P1
Application
Critical Master
Router
P2
Application
Critical Shadow #1
P3
Application
Critical Shadow #2
Multicasted Critical Update Msg.
Multicasted Critical Update Cfm.

Figure 66

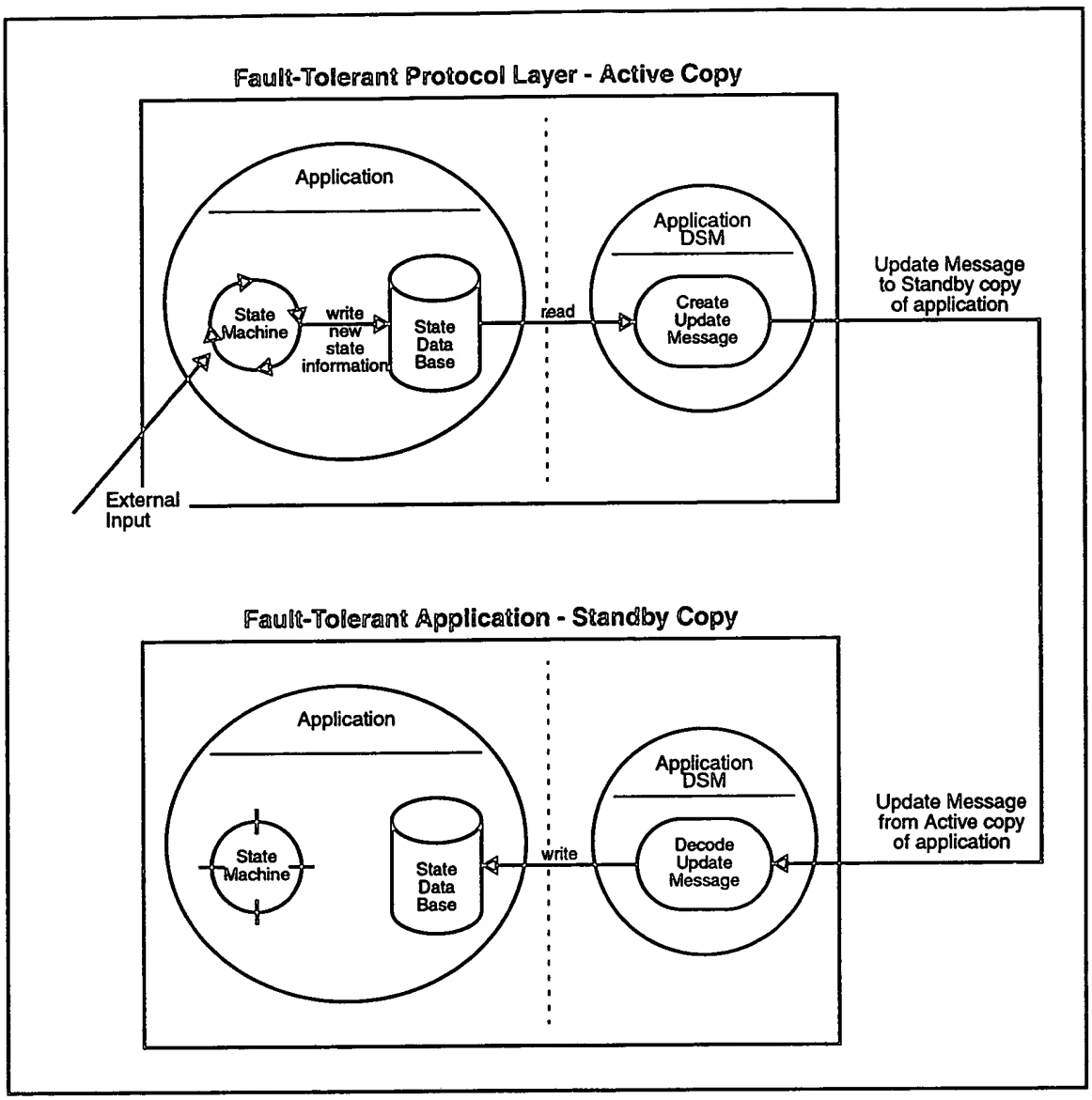
Fault-Tolerant Protocol Layer - Active Copy
Application
State Machine
write new state information
State Data Base
read
Application DSM
Create Update Message
Update Message to Standby copy of application
External Input
Fault-Tolerant Application - Standby Copy
Application
State Machine
State Data Base
write
Application DSM
Decode Update Message
Update Message from Active copy of application

Figure 67

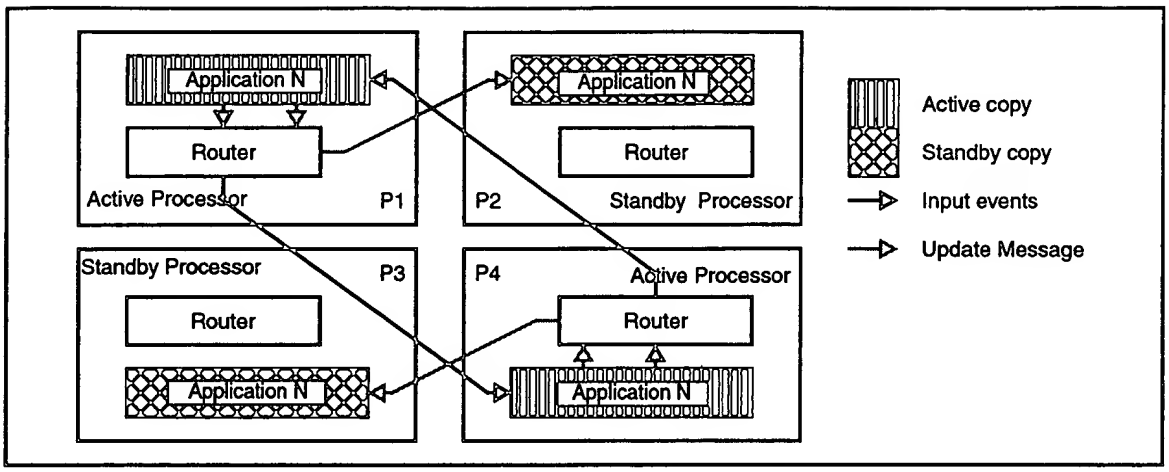
Application N
Router
Active Processor
P1
Application N
Router
P2
Standby Processor
Standby Processor
P3
Router
Application N
P4
Active Processor
Router
Application N
Active copy
Standby copy
Input events
Update Message

Figure 68